Pablo Rodríguez

Un solo origen

Un método automatizado de edición electrónica

http://www.un-solo-origen.tk

Este documento se ha generado con
`pandoc` (`http://pandoc.org/`).

La composición tipográfica del documento
PDF se ha realizado usando ConTEXt
(`http://contextgarden.net/`).

Las tipografías empleadas en el documento
PDF para impresión en papel son *TEX Gyre
Pagella*, *GFS Didot*, *URW Classico* y *Mechanica*.
El documento ePub para dispositivos
electrónicos contiene la tipografía *Tempestiva*.

Un solo origen

Un método automatizado de edición electrónica

Pablo Rodríguez

2015

http://www.un-solo-origen.tk

© 2015 Pablo Rodríguez (`http://www.un-solo-origen.tk`). Todos los derechos reservados.

El Centro Español de Estudios Repográficos —CEDRO— carece de autorización para percibir derechos de ninguna naturaleza por este libro. Esta obra no figura en su catálogo.

No son objeto de copia privada «las efectuadas en establecimientos dedicados a la realización de reproducciones para el público, o que tengan a disposición del público los equipos, aparatos y materiales para su realización» (Real Decreto 1657/2012, artículo 3.4.a), `https://www.boe.es/buscar/act.php?id=BOE-A-2012-14904#a3`).

23 «Autómata» proviene del adjetivo plural αὐτόματα, que tomado como nombre significa «ingenios mecánicos», pero como adjetivo es «que obran por sí mismos». Curiosamente, la palabra inglesa *robot* viene del checo *robota*, que es el trabajo gratuito debido al señor feudal.

24 Ley de Propiedad Intelectual, artículo 66, en `https://www.boe.es/buscar/act.php?id=BOE-A-1996-8930#a66`.

25 «Mecanografía» puede retrotraerse a μηχανῇ γραφή. Literalmente significa «escritura con máquina».

26 Tiene un procesador *Intel i7-7500U*, con 8GB de memoria de trabajo y disco de estado sólido. Como sistema operativo, está instalado *Fedora Linux 28 LXDE Edition*.

27 Se trata de una hoja DIN A4 que contiene dos folletos en tamaño DIN A5, que han de cortarse por la mitad.

28 El filtro de `pandoc` para eliminar las notas ralentiza la generación. Es lógico que sea más lento si tiene que hacer más cosas.

10 Por supuesto, se trata de una media. Puede ser que en algún caso sea más lento. O también más rápido.

11 *Fedora Linux 22*, que puede encontrarse en `https://getfedora.org`.

12 Que `pandoc` o ConTEXt están disponibles para cualquier sistema operativo no significa que podamos instalar una versión de un sistema operativo distinto. Cada sistema operativo necesita que los programas que se instalen sean para ese sistema.

13 El tamaño DIN A4 tiene las medidas de 210 × 297 mm. El tamaño carta mide 8.5 × 11 in. Si una pulgada son 2.54 centímetros, la carta mediría 21.59 × 27.94 cm.

14 El modo de hacerlo es muy fácil. Se copia el archivo que queramos y cambiamos la extensión `.docx` u `.odt` por `.zip`. Luego abrimos el archivo comprimido con el descompresor adecuado. Debe hacerse siempre con una copia y nunca con el archivo original, por seguridad básica.

15 La segunda o tercera versión de la *GNU General Public License*. Por si sirve de referencia, la segunda versión de esta licencia es con la que se publica *Linux*.

16 A. Machado, *Proverbios y cantares*, en *Nuevas canciones*, LXVIII.

17 O para una copistería, que es la versión de la imprenta disponible para el resto de los mortales.

18 «πάθει μάθος»; Esquilo, *Agamenón*, 177.

19 «τὸ δὲ μέγιστον, ὅπερ εἴπομεν ὀλίγον ἔμπροσθεν, ὅτι δυοῖν ὄντοιν, τοῦ τε ὄντος καὶ τοῦ ποιοῦ τινος, οὐ τὸ ποιόν τι, τὸ δὲ τί, ζητούσης εἰδέναι τῆς ψυχῆς»; Platón, *Carta VII*, 343B–C.

20 La generación de un documento PDF es en buena parte la impresión a un archivo digital.

21 En realidad, hay referencias cruzadas en los índices o en las notas al pie. Si bien, ahí las referencias y los identificadores únicos los genera automáticamente el ordenador.

22 No me gusta escribir *tablets*. Es un barbarismo.

Dedicado a mis padres, cuyo amor y cariño constantes me han permitido ser.

NOTAS

1 D. Knuth, *The TEXbook*, Addison–Wesley, p. 9.

2 πᾶν es «todo» en griego y *doc* está por *document*.

3 Creo que en esa funcionalidad, *Calibre* no es tan valioso. O es mucho más lento que copiar los libros directamente. Además, así llena los archivos de datos innecesarios.

Siendo *Calibre* un buen programa, en mi opinión, creo que como gestor de libros electrónicos no merece la pena. Aunque es sólo mi opinión y es independiente de la edición digital.

4 Si escribimos una orden, invocando `pan-doc` en vez de `pandoc`, el sistema operativo nos mandará un mensaje de error porque el programa no existe. Cito la posibilidad de una partición silábica distinta a la predeterminada como anécdota de una mayor capacidad.

5 Un ejemplo de generación combinada de documentos sería generar presupuestos, contratos y facturas partiendo de los mismos datos. Y en una misma operación, no en tres operaciones distintas.

6 En realidad, hay que añadir que de los cuatro documentos PDF se generan otros cuatro para impresión doble directa: folleto doble en DIN A4, con notas y sin notas, y folleto doble en carta, con notas y sin ellas.

7 Para quien le pueda interesar, tiene un procesador de doble núcleo a 1.6 GHz y 2 GB de memoria de trabajo.

8 Es algo que está en proyecto, si bien no se ha iniciado. El día que esa característica esté disponible, de los quince segundos en mi ordenador pasaremos a cinco segundos.

9 Por citar algunos aspectos: qué caracteres va a tener en cada línea, cómo va a partir las palabras que no quepan en una línea, cómo va a partir las páginas, calcular que no queden líneas viudas y huérfanas. Y ésos sólo son algunos ejemplos.

Computers are good at following instructions,
but not at reading your mind.[1]

A EXPERIMENTANDO DE NUEVO

Después de cambiar de ordenador tras una docena de años, mi nuevo portátil[26] tarda cincuenta segundos en la operación completa de generación desde un único archivo de origen —como he explicado en *II. El experimento inicial*—, del conjunto de:

- Documento ePub, con y sin notas al final —generados con `pandoc`—.
- Documento MOBI, con y sin notas al final —convertidos de los anteriores por *Calibre*—.
- Documento PDF de alta calidad, con y sin notas, para tamaños de papel DIN A4 y carta americana —generados por ConTEXt—.

  De los cuatro documentos PDF, se generan también otros cuatro documentos dispuestos para impresión en folleto duplicado en la hoja[27].

El documento ePub tarda menos de medio segundo y menos de un segundo —según sea con notas o sin ellas[28]—. *Calibre* tarda casi tres minutos y medio en convertir cada uno de los documentos ePub a formato MOBI.

ConTEXt tarda siete segundos en componer un documento PDF completo. Necesita casi un segundo para generar el folleto sencillo y dos segundos para duplicar el folleto en la hoja completa que corresponda —DIN A4 o carta—.

nos será mucho más fácil no olvidar. Afianzaremos las raíces del saber en nuestra mente.

Al llegar a este punto, nos puede entrar un mayor o menor temor. Nos susurra al oído que este aprendizaje está muy por encima de nestras capacidades. Me sorprendería muchísimo que así fuese. Nos sobran las capacidades. En mucha mayor medida de lo que pensamos. Hablar una lengua —poder leer y escribir en esa lengua— es una capacidad que exige facultades mucho mayores.

Lo que falla no son los ordenadores. Tampoco el problema está en la informática. La cuestión está en cómo vemos la informática. Cómo pensamos que son los ordenadores y los programas. Eso se debe a malas explicaciones. Y malos entendidos. A no buscar comprender cómo funciona la máquina. Con corregir eso, tenemos ya mucho ganado.

Únicamente nos queda aprender a hacer. Aunque sabiendo que estamos mandado a un ayudante de despacho. Este asistente va a editar nuestros escritos. Sólo con que se lo mandemos correctamente. Las órdenes concretas las explico en *Aprender* `pandoc`. Ahí cuento lo que hay que aprender en la práctica.

CONTENIDO

Sobre la licencia de uso 11

Introducción 13

I Los programas 17
A Ordenadores, no programas B Edición digital de textos

II El experimento inicial 22
A El resultado final B ¿Es muy lento? C Un proceso automático

III En donde quieras 26

IV ¿Por qué no *Word*? 28
A El procesador de textos B El programa C Lo mejor de cada casa

V El coste real 37
A Valor B Coste C Aprendizaje D Esfuerzo

VI La simulación visual 45
A Una metáfora de actividad B Imágenes y palabras C Inmediatez D Recursos

VII Qué es, no cómo está 53
A Una buena referencia B Elementos C Atributos D Propiedades

VIII Habla con ella 62
A Un nuevo contexto B La máquina C La comunicación D La automatización E El asistente

IX Mecanografiado 69
A La composición B La tarea C El salto digital D Nuevas posibilidades

Conclusiones 75

A Experimentando de nuevo 77

Notas 79

CONCLUSIONES

Estas páginas han tratado de mostrar la nueva perspectiva necesaria para la edición digital. En cierta manera, hay que dejar de lado la imagen del procesador de textos. Aunque esa pérdida se produce por una ganancia mejor. La de un asistente que es capaz de editar directamente a partir de un único manuscrito.

La ganancia se produce en un mejor modo de trabajo con textos. Se consigue más, con mejores resultados y en menos tiempo. Quizá lo que mejor defina el método que describo es mucho menos engorro. Tanto a la hora de escribir, como a la hora de editar.

Por supuesto, hemos ganado una nueva perspectiva. Teórica, si se quiere. Vemos ciertas cosas desde otra perspectiva. Aunque está claro que se trata de una práctica. Y de una práctica que hemos de aprender. Por supuesto, si queremos. Si consideramos que nos merece la pena. Y aquí tenemos que decidir cada cual.

La única exigencia previa y básica es querer aprender. No me refiero sólo a la disposición de esforzarse. No sé por qué me parece una preparación al sufrimiento muy difuminada. Querer aprender supone una voluntad clara de buscar entender. De no memorizar una catedral de informaciones, que antes o después se nos indigestaría.

Si no buscamos entender, nos imponemos el aprendizaje como una pesada carga. No siempre entenderemos todo en el primer minuto. No debemos dejar de aprender por ese motivo. A la larga, o entendemos o nos perderemos. Además de que

muchos errores y con buenas intenciones se corrigieron otros que no eran tales. Con el sistema adecuado, podrían haberse evitado esos fallos.

Los sistemas de control de versiones son especialmente útiles donde es decisivo conocer con detalle cada uno de los cambios. Uno de los sistemas más conocido es `git`, desarrollado precisamente para el núcleo de *Linux*. Pongámonos en situación. *Linux* tiene un modelo de desarrollo abierto. Cualquiera puede aportar código. Si el código tiene más de doce millones de líneas, es básico poder controlar los cambios. Por la más mínima seguridad. Porque que algo no falle en un determinado momento, no quiere decir que no vaya a fallar nunca.

Evidentemente, ningún libro será tan voluminoso. Ni siquiera una enciclopedia. Sin embargo, la seguridad de saber cuándo o cómo ha cambiado con exactitud es algo impagable. No soy programador. Mi formación es en humanidades. Escribo mis textos solo. Y considero que usar un sistema de control de versiones es una de mis mejores decisiones. Porque hay muchos detalles que puedes dejar que los haga el asistente. Y dedicar la cabeza a otras cosas. Por ejemplo, a escribir.

SOBRE LA LICENCIA DE USO

Este libro se ofrece con descarga gratuita en `http://www.un-solo-origen.tk`. Esta obra no está bajo una licencia *Creative Commons* u otra licencia libre similar.

Mi intención es poner este libro a disposición de toda persona que quiera leerlo. Espero que sea útil. A cambio, sólo pido un sencillo pacto de nobleza. Creo que cualquiera estará de acuerdo.

- Si el libro te parece útil, recomiéndalo a quien le pueda interesar. Hazle llegar la dirección `http://www.un-solo-origen.tk`. Así tendrá la última versión disponible. También sabré el número de descargas con bastante fiabilidad.

- Si consideras que tiene puntos mejorables, no dejes de decir cuáles son. Tampoco olvides de describir en qué deben mejorar.

- Si consideras que el libro es malo, agradezco la crítica. La única condición es que sea respetuosa con las personas.

Los comentarios son incidencias en `https://github.com/ousia/un-solo-origen/issues/`. Tienes que estar inscrito en *GitHub*. Todos los comentarios son públicos.

Para que no haya incomprensiones, hago las aclaraciones pertinentes para que nadie se lleve a engaño.

Únicamente se permite la descarga desde la página mencionada. Se permite el uso personal —no colectivo— de los

archivos. Si quieres compartir el libro, envía `http://www.un-solo-origen.tk`.

Dentro del uso personal, se incluye la impresión del archivo PDF en establecimiento público. En la página se explica cómo hacerlo. El establecimiento no está autorizado a vender el libro, sólo a imprimirlo a petición de la persona interesada. El precio de impresión y encuadernado no podrá ser superior al del mismo número de páginas y encuadernación de cualquier otro documento.

Una última aclaración para las entidades de gestión colectiva de derechos —Centro Español de Derechos Reprográficos, CEDRO—: no cabe el cobro de licencia por fotocopia de esta obra. Este título no se encuentra dentro de su catálogo.

No existe copia privada en fotocopiadoras de uso público por excepción reglamentaria (Real Decreto 1657/2012, artículo 3.4.a), `https://www.boe.es/buscar/act.php?id=BOE-A-2012-14904#a3`).

Por tanto, no pueden cobrar por esta obra. Ni por copia privada, ni por cualquier acuerdo o programa de licencias con su entidad —`https://www.conlicencia.com`—. Ustedes carecen de ella.

elementos. Para escribir sólo hay que teclear. No hay que hacer operaciones extrañas al mismo hecho de pulsar el teclado. Así, escribiendo estamos únicamente centrados en el hecho de escribir. En otro momento nos dedicaremos a componer tipográficamente el texto.

La edición del texto supone dar las instrucciones para que el ordenador haga todo el trabajo de composición tipográfica. Por supuesto, no me refiero a la edición literaria del texto —como la corrección de errores—. Nuestro asistente de impresión es un gran matemático. De ahí que el ordenador pueda ser un excelente cajista digital. Se trata de automatizar la composición tipográfica del texto. Y en diferentes formatos la elaboración del texto digital será ligeramente distinta. El ordenador puede seguir instrucciones, pero no es autónomo. Nosotros le damos las pautas para que componga nuevos archivos de único archivo origen, según nuestras propias necesidades.

D NUEVAS POSIBILIDADES

Poder trabajar con textos digitales sólo teniendo que mecanografiar tiene ventajas adicionales. No se trata sólo de los resultados o la tarea misma. Son ventajas que existen por trabajar con un mecanografiado que es totalmente transparente. Probablemente existen muchas posibilidades. Sin embargo, quiero describir brevemente una que considero que es especialmente útil. Es el control de versiones.

Dependiendo mucho del modo en que trabajemos, es muy fácil que necesitemos saber cuándo, qué y quién ha cambiado algo. Eso supone tener un control de versiones de los textos. Sin la informática, es casi una tarea imposible. Como curiosidad, ése es el grandísimo problema que tiene *The Lord of the Rings*. Me refiero exclusivamente a la versión original salida de la pluma de Tolkien. En diferentes ediciones, se introdujeron

compone. De otro modo no podría mostrarlo en pantalla. Ahora bien, la calidad no debe ser buena, cuando las editoriales no usan *Microsoft Word* para editar textos.

La imagen visual de los procesadores de textos nos puede hacer pensar que escribir y editar un texto no son tareas separadas. La misma conclusión podemos sacar con programas como *Adobe InDesign*. Parecería que tuviésemos que tener el texto siempre compuesto. Que se trata de tareas irremediablemente unidas. No separar escritura y edición es la manera más fácil de trabajar con distracciones en ambos casos.

Es cierto que nuestro asistente digital nos permitirá escribir el texto. Este mecanografiado será igual que el que se enviaba hace medio siglo a cualquier imprenta. También el asistente digital puede hacer las funciones de cajista digital. Sólo hay que usar el programa adecuado. En diferentes formatos, con diferentes reglas. Lo importante aquí es que el texto mecanografiado será mutuamente inteligible y totalmente visible para persona y máquina.

La persona escribirá el texto con todas las indicaciones a la vista. Por ejemplo, qué es un título o qué es una nota al pie. En el manuscrito sólo estará qué es cada cosa. Eso es escribir, redactar el texto. En otro momento, indicaremos a nuestro asistente de impresión cómo tiene que componer cada cosa. Por ejemplo, las páginas, cada tipo de título o las notas al pie. No se trata necesariamente de un momento posterior a la escritura. Aunque es un momento distinto. Porque realmente es una actividad distinta.

Separar el texto mismo de la presentación nos permite trabajar mejor, más rápido y con mejores resultados. Aunque sea otra persona la que haya de editar digitalmente el texto. Hay una transparencia total y absoluta respecto al texto y a sus

INTRODUCCIÓN

Estas páginas cuentan una historia. Relato un ejemplo de edición digital de textos. Creo que las características que hacen relevante este relato son tres. Por esas particularidades, quizá compense leer estas páginas. Las tres peculiaridades están relacionadas entre sí.

La primera característica es que la edición se realiza en diferentes formatos. Mejor, en formatos de edición digital y de edición en papel. La segunda propiedad es que se generan de un solo archivo de origen. De un único y el mismo formato común sale todo. La tercera funcionalidad es que la edición es totalmente automatizada. Pulsando una tecla, se generan los archivos necesarios. Por supuesto, todo con calidad profesional.

Antes de seguir —casi antes de empezar—, conviene aclarar algo. El método que aquí describo creo que puede ser útil. Desde luego, para mí lo es. Cada persona tiene que valorar si tiene utilidad en su caso particular. En el mío, no pretendo de ningún modo vender nada. Ofrezco una historia que entiendo que puede interesar a otras personas. Absolutamente nada más.

Volviendo al texto, como ya he dicho, contiene un relato. Principalmente porque no se trata de una explicación técnica. La explicación técnica será propia de manuales o escritos técnicos. Algunos estoy escribiéndolos. Otros espero escribirlos pronto. En todo caso, este texto ni es, ni pretende, ni puede ser un manual. Aunque pueda contener alguna indicación más o menos técnica a modo de ilustración.

Lo que quiero dar en estas páginas es la visión más general posible. Es un poco saber qué hay, con qué podemos contar. Sobre todo para poder examinar, antes de ponernos manos a la obra, si realmente nos interesa o no. Creo que se trata de una buena alternativa de edición digital. Diría que es la mejor que conozco. Si alguien tiene una mejor, le pediría que nos la cuente a los demás.

En el relato que sigue en estas páginas hay tres elementos: hechos, reflexiones y programas. Los hechos —o más bien serían resultados— por sí mismos darían para muy poco. En una página estarían expuestos. Con esa exposición de resultados, yo mismo exclamaría: «¡ah!, pues muy bien». Porque lo que pueda haber conseguido, en el mejor de los casos puede suscitar la curiosidad ajena. Pero nada más.

Las reflexiones son necesarias para entender qué se puede conseguir. Tanto por mi parte, en lo que el método automatizado de edición digital puede ofrecer. Como por quien entienda, para saber qué puede conseguir de unificar todo en un solo origen. Ahí habrá datos técnicos. Procuraré hacerlos máximamente inteligibles. No hay que memorizarlos, sólo situarlos como partes del relato. De igual modo que un detalle en una novela: lo esencial no es retenerlo, sino integrarlo en el relato que el texto cuenta.

Los programas son personajes de este relato, aunque trataré que no tengan un protagonismo excesivo. Porque lo que trato de contar no son instrucciones informáticas, sino ver cómo realiza la edición digital de textos. Es lo mismo que con las consideraciones e informaciones técnicas: no se trata de aprender de memoria, sino de entender qué puesto ocupan en la imagen global.

B LA TAREA

La tarea de la edición en la imprenta tradicional está claramente diferenciada de la redacción del texto. Tanto en tiempo, como en personas. Una persona es la que escribe el texto. Otras lo editarán. Para que esa cadena funcione, el texto tiene que estar totalmente terminado. De hecho, hasta la propia ley reconoce que durante la corrección de las pruebas se admitirán sólo aquellas modificaciones consideradas imprescindibles[24]. Referido a las obras impresas, es imposible componer un texto si no está fijado de modo definitivo.

Esa separación estricta de tareas se nos hace extraña hoy. O no es absolutamente necesaria. Si bien podemos editar para publicar y vender, existen otras posibilidades y otras necesidades. La diferencia con la imprenta de hace un siglo es que conocía su oficio. La introducción de la informática ha desorientado a quienes escriben y a quienes editan. Porque saber usar programas de edición no quiere decir siempre saber editar.

C EL SALTO DIGITAL

Es posible que hace un siglo, las imprentas admitiesen manuscritos. Posteriormente, preferirían textos mecanografiados. A día de hoy, nadie admite textos manuscritos o mecanografiados. Por supuesto, seguro que habrá excepciones. Aunque probablemente tendrá que tratarse de textos realmente extraordinarios. Además, ha de coincidir también que antes acepten leerlos si no están en soporte informático.

El cambio digital supone varias cuestiones. La nueva escritura a máquina es ya con ordenador[25]. En el mismo ordenador ha de hacerse la edición del texto. De hecho, un procesador de textos compone el texto. De una manera pobre, aunque lo

estilo, sólo ha de notarse por su ausencia o su mala calidad. Por eso, aunque nos cueste entenderlo, la tipografía no está sometida a modas pasajeras. Ése sería el dictado para el diseño gráfico. En la buena tipografía, el libro que sale de la imprenta tiene una composición intemporal. La idea básica es que sea legible, no que quede bonito.

La legibilidad es la finalidad de la composición tipográfica. Nos puede parecer extraño, pero ha habido una bajada de calidad tipográfica desde el uso de los ordenadores para la edición. No todo es malo, pero si nos acercamos a una librería encontraremos demasiados ejemplos. De libros que son ilegibles por la tipografía, el tamaño o el interlineado. Otros libros tienen márgenes o dimensiones de página difíciles. En los títulos que haya pasajes en otros alfabetos, hay soluciones sencillamente pésimas, auténticos crímenes de lesa cultura. O puede tratarse de obras escritas totalmente en español, no especialmente mal compuestas a primera vista, pero con resultado duro cuando se llevan leídas más de un centenar de páginas.

La composición tipográfica de la imprenta tiene que ver con la composición de cajas. Los tipos son metálicos y se trata de crear bloques mediante la unión de unos tipos con otros. Eso se entintaba y se imprimía sobre el papel. De ahí que un oficio en la imprenta fuese el de cajista. No se trataba de poner letras sin más, sino que había que añadirles los espacios necesarios. Y ajustarlos lo más posible para formar líneas, párrafos y luego páginas. Eran cajas de madera, que se atornillaban. Se hacía todo a mano, manchándose de la tinta medio seca del uso anterior. Se trataba de una labor que no sólo requería experiencia, sino que exigía maestría.

Como última cuestión, espero que este texto sea útil. Para cualquier comentario relacionado con el tema, abre una incidencia en `https://github.com/ousia/un-solo-origen/issues/`. Requiere inscripción en *GitHub*. No olvides que todos los comentarios son públicos.

IX MECANOGRAFIADO

El sentido de la edición es cómo componer tipográficamente un texto. Además de publicar, editar es sacar un libro de un texto simplemente escrito. En nuestro caso, la edición digital es la composición tipográfica de un texto con medios informáticos. Y el texto escrito no es sólo manuscrito, sino también mecanografiado.

Para entender qué es lo que supone la edición digital, es útil que nos retrotraigamos a comienzos del siglo pasado. Más o menos, cien años. No pretendo decir que cualquier tiempo pasado fue mejor. Es sencillamente para poder observar cuál es la referencia de la edición en un mundo en el que no existen ordenadores en absoluto. Esto es importante para ver qué es la edición sin más. Sin visiones lastradas por la informática personal. Es la visión del auténtico artesano, de quien conoce los fundamentos de su labor.

A LA COMPOSICIÓN

La composición tipográfica es el paso previo a la impresión. Para eso se parte de un texto escrito y de la maqueta de la página. Esto es, primeramente se diseña cómo ha de ser la página en que se quiere componer el texto. Sin unos límites previos —aunque no lo entendamos por la informática—, es imposible dar forma alguna a ningún texto.

La edición del texto tiene la finalidad de hacerlo lo más legible posible. La composición tipográfica no es diseño gráfico. No se trata de hacer un texto vistoso. El cometido de la buena tipografía es hacer que entre por los ojos. Como todo buen

Así sólo tenemos que contemplar el resultado de la ejecución de la tarea ordenada.

Este asistente se asemeja a lo que en el estamento aristocrático se llamaba el ayudante de cámara. Bien, un ordenador es nuestro siervo. Y nos ayuda en tareas de despacho. Copia nuestros textos, envía nuestros mensajes, nos hace una selección de prensa… Los ejemplos podrían ser infinitos, porque las tareas en las que nos asiste son virtualmente infinitas. De ahí que el ordenador pueda verse como nuestro asistente de despacho. No se dedicará a nuestra vestimenta o aposento —para eso necesitamos un autómata completo—, pero gestiona todos nuestros documentos.

En estas páginas pretendo mostrar una ámbito especial en que su ayuda es especialmente valiosa. Es por lo que es nuestro asistente de impresión. No es que nos gestione la impresora. Impresión tiene que ver con hacer presión sobre. Exactamente, sobre la hoja de papel. La presión sobre la hoja de papel la lleva a cabo la imprenta. Nuestro asistente de impresión es capaz de generar automáticamente la composición tipográfica a partir del texto. En papel o en formato digital. Todo eso, sin mancharnos las manos de tinta.

I LOS PROGRAMAS

En las obras de teatro clásicas se incluía antes del texto un capítulo con el título de *Dramatis personae*. Viene a significar «personajes de [esta] historia». Es muy útil para orientarse con quién es quién a medida que avanza el relato. En este capítulo pretendo hacer precisamente eso: ver qué programas son los que propongo para la edición digital automatizada.

La introducción de los personajes de esta historia quedaría incompleta si sólo los nombrase. Para hacerlos más amables, es importante describirlos, mostrar cuáles son sus principales características. Esa presentación será breve. Para que nos pongamos en situación, imaginemos que nos presentan a alguien. Puede ser que esa persona nos haga ser lo más extrovertido del planeta. Pero creo que la mayoría preferiremos que además de darnos su nombre, nos den más detalles. No por curiosidad o aumentar nuestra información respecto a terceros. Es sencillamente para facilitar la conversación posterior.

No pretendo abrumar a nadie con datos. Con la imagen del párrafo anterior, digamos que se trata de gente que nos puede interesar conocer. No hay que memorizar todos y cada uno de los detalles, sino ver que son datos de algo unitario. Y lo que en nuestra cabecita aparece como disperso, se irá juntando conforme vayamos sabiendo cada vez más. Ése es el motivo por el que este primer capítulo tiene sentido.

A ORDENADORES, NO PROGRAMAS

Antes de las presentaciones, creo que nos ahorraremos bochornos innecesarios si entendemos algo. Se trata de algo muy

básico, pero que nos impide sacar de la informática todo su potencial. Y no se trata tanto de una cuestión de conocimientos, como de perspectiva. No se trata de saber programar, sino de entender qué es y cómo funciona un ordenador.

No es un conocimiento profudísimo, sino algo extremadamente sencillo. Los ordenadores son máquinas universales. Los programas son herramientas que usamos con y en esas máquinas. Si bien ambas son necesarias, es más importante saber manejar un ordenador que conocer un programa. Por una razón fundamental: se trata de entender qué hacemos. Sólo así podremos saber qué hemos de realizar y qué podemos llevar a cabo.

Con algo que veremos más adelante, *Microsoft Word* no es un estándar. Es sólo procesador de textos de una determinada marca. No es cuestión de que sea mejor o peor. Es que es sólo un modo de hacer las cosas. Si sólo usamos *Microsoft Word* —o *LibreOffice Writer*, para lo que trato de explicar es lo mismo—, no sabemos trabajar con textos. Únicamente somos capaces de usar un procesador de textos determinado. En cuanto cambiasen las circunstancias, nos estrellaríamos. Porque no sabemos realmente qué estamos haciendo.

En el ejemplo que comento, lo que está en juego es saber qué hace un procesador de textos con el texto digital. Cómo lo procesa por dentro. Qué nos oculta. No se trata de desconfianza, por supuesto. Pero si nos dan purés para niños, el alimento sólido desaparecerá ante nuestros ojos. Con la papilla cabe engullir, o desde luego no es posible masticar. Si el procesador de textos es sólo una opción, quizá podamos encontrar mejores. Porque sean más efectivas, porque logremos más con menos esfuerzos. Aunque hay que aprender a masticar, a sacar los dientes, a trabajar de otro modo.

La máquina trabaja sola, aunque requiera supervisión. La herramienta es sólo un utensilio empleado en la actividad propia. Un ordenador es una máquina, los programas son herramientas. El ordenador no va a trabajar por sí mismo, también necesita supervisión. Existen muchas tareas que son mecánicas y no genuinamente humanas. Aunque las realicemos las personas, tenemos que aprender a mandar para que las hagan las máquinas.

Para saber mandar hay que saber qué es lo que se ordena. En el caso de una calculadora binaria, no sólo hay que saber comunicar mandatos para su ejecución. Es necesario saber qué puede hacer un ordenador y cómo lo hace. Ese saber corresponde al conocimiento de la informática. Saber manejar un ordenador supone plantear la tarea para que toda o la mayor parte pueda hacerse automáticamente. La eficacia del ordenador está en nuestra capacidad de mandar a una máquina extraordiariamente rápida y altamente eficiente. Y la orden no es más que establecer el contexto para que el ordenador haga la tarea sin necesidad de nuestra intervención.

E EL ASISTENTE

Un ordenador es una máquina con la que podemos automatizar la ejecución tareas a alta velocidad. El ordenador trabaja por nosotros, lo hace mucho más rápido y sin errores. Nuestro puesto es el de mando. De este modo, lo que tenemos que entender es que el ordenador es un asistente. Este ayudante ejecuta órdenes complejas por sí mismo. Es altamente eficaz, porque es preciso y más rápido de lo que somos la mayoría de personas. Sin embargo, ni sabe, ni entiende nada. Por tanto, no aprende nada. Somos las personas las que tenemos que aprender a darle órdenes cada vez más útiles, más completas.

sajes completos es posible formular un mandato y evitamos saltar de inmediatez en inmediatez.

Las interacciones son la selección de lo que está a la vista. Ésa es precisamente la grandeza y el límite de la aproximación visual. Mediante órdenes escritas se pueden articular mandatos complejos. Dicho de un modo totalmente simple, así logramos hacer lo que de otro modo no podríamos o nos llevaría más o mucho más tiempo. Cuando sabemos mandar, cuando comunicamos con el ordenador en sus términos, es cuando el ordenador empieza a hacer todo el trabajo para nosotros. De este modo, el ordenador obedece y la persona sólo tiene que mandar.

D LA AUTOMATIZACIÓN

Todo el sentido de que la comunicación con el ordenador sea lógica es ordenar la ejecución de tareas a la máquina. En un modelo de interacción visual, la máquina realiza tareas simples. Como ya he explicado, la imagen del escritorio es una metáfora visual de la tarea humana. Es un compromiso entre la facilidad de uso y las posibilidades operativas del ordenador. En ciertos contextos, es extremadamente útil que sea así. Aunque en muchos casos se queda corto.

La tarea propia de un ordenador es la automatización de tareas. Autómata es sinónimo de robot[23]. En ambos casos, a lo que se refieren ambos nombres es a lo que obra por sí mismo. Por tanto, automatismo no es lo que se hace solo, sino lo que no necesita intervención externa para que se lleve a cabo.

El automatismo es una actividad que se llega a su término sin mediación externa. Precisamente eso es la acción de una máquina, tras el mandato de una orden. Y ahí está la diferencia entre que el ordenador sea una máquina o una herramienta.

Que las cosas sean distintas de lo que imaginábamos no es un mal signo. Es un síntoma absolutamente esperanzador. Es muy probable que estemos a punto de aprender algo. Por supuesto que cuesta ir contra incomprensiones anteriores propias. Nos cuesta deshacernos de ellas porque son nuestras. Sin embargo, si aprendemos, adquiremos conocimientos y capacidades mejores.

No me olvido que este libro no debe contener instrucciones de manejo de ordenadores o programas. Como ya he dicho, eso será para otros textos. Ahora bien, muestro la necesidad de ampliar el horizonte. Porque sin ese horizonte nos perderemos muchas cosas. Quizá lo que cuenta este libro es sólo un pequeño ejemplo de eso que nos estamos perdiendo.

B EDICIÓN DIGITAL DE TEXTOS

Estas páginas tratan de lo que describo como un método automatizado de edición digital. Si se quiere, puede abreviarse a edición automatizada de textos. Con ambas expresiones, lo que trato de describir es que pulsando una tecla, se consigue todo lo necesario.

Por supuesto, la informática no es magia. Al menos, yo no soy informático. Ni tampoco mago. Para que con una tecla funcione todo, tenemos que tener antes unos elementos previos. Son esencialmente el texto y los programas. Del texto entiendo que no es necesario que hable ahora.

Los programas para edición de textos son tres. Como prometí al principio del capítulo, los nombro y los presento brevemente. Aunque no lo pretendo, cada uno tiene un modo distinto de presentación. De momento, dejemos a un lado las razones del diferente formato. Los programas son `pandoc`, *Calibre* y ConTEXt.

1 PANDOC

Como su propio nombre indica[2], `pandoc` es un conversor universal entre distintos formatos de texto. Es extraordinariamente rápido en sus conversiones.

`pandoc` es una herramienta excelente, aunque su uso exija un cierto conocimiento. Porque si bien convierte entre muchos formatos, existen límites. Porque hay formatos de los que no puede leer, o no escribir, o ni leer ni escribir. Y también porque hay documentos que tienen una conversión muy difícil. Porque en el fondo, no están bien estructurados —hablaremos de eso más adelante—.

2 *CALIBRE*

Calibre es programa mastodóntico. Debido a su inmensidad, existen tareas que realiza más lentamente que, por ejemplo, `pandoc`. Es capaz de convertir entre bastantes formatos de libros digitales. En mi opinión, su conversión no es muy limpia. Añade demasiados datos innecesarios, al menos si lo comparamos con `pandoc`.

Tiene un visor de libros electrónicos que es muy bueno. También realiza conversiones a formatos que `pandoc` no es capaz de hacer de momento. Es también un gestor de libros electrónicos, que los administra entre el ordenador y el lector digital[3].

3 CONTEXT

ConTEXt es un programa de generación de documentos PDF de alta calidad tipográfica. Es lo que en la imprenta clásica se llama un cajista digital. Ofrece la posibilidad de conseguir resultados de imprenta. Igual que con la imprenta clásica

a la ironía: con tanta inteligencia en el ambiente, no sé si las personas nos volveremos profundamente idiotas.

Bromas aparte, la nueva informática impone un paradigma. Entre otras cuestiones, la informática parece más asunto de formas y colores que de ceros y unos. En los dispositivos móviles, nuestros dedos son el ratón. Son el puntero, que señalan por contacto. Por supuesto, no estoy en contra de que existan entornos visuales. El único problema es que eso sea toda la informática. O nuestro modo completo de manejar ordenadores. Principalmente, porque nos estaremos perdiendo lo más importante.

Los ordenadores son máquinas complejas, aunque en el fondo sean profundamente sencillas. Si tratamos de hacer simple lo que en realidad es complejo, caben dos posibilidades. O el resultado es una máquina simplona, o extraordinariamente complicada. No podemos negar que los susodichos teléfonos inteligentes son ordenadores muy sencillos. Y teléfonos extraordinariamente complicados —no se puede tener todo en esta vida—. Pero como ordenadores, presentan bastantes limitaciones. Podremos disentir, pero no conozco a nadie que haya sustituido su ordenador —en casa o en el trabajo— por un teléfono inteligente. Y no creo que sea sólo por el teclado o la pantalla.

El modelo de interacción del nuevo paradigma es el contacto. No la persona, sino la pulsación. Para un dispositivo móvil es suficiente. Sin embargo, es un modo precario de comunicación informática. Esa interacción yo diría que no es propiamente comunicación. Porque el contenido de las órdenes es muy pobre. Están limitadas a lo inmediato. La comunicación articulada permite superar la inmediatez del «esto». Con men-

calculadora. De ahí que sea fría. En realidad es totalmente desapasionada. Mantener la calma puede ser una buena manera de seguirle el juego. Su contumacia se debe a que su eficacia ejecutora se basa en estricta comunicación lógica, no en la empatía.

La comunicación lógica tiene que ver con el uso de palabras. No está relacionado con sentimientos, ni hay que mudarse a lugar alguno. Las palabras permiten construcciones básicas. Esa lógica es como un puerto de comunicación. Pensemos en un puerto universal de serie, también conocido como USB. La complementariedad es perfecta y permite una comunicación muy rápida, porque hay una estructura común. Esa lógica es la que permite a persona y a máquina comunicar.

La comunicación lógica de la que hablo no es conversación en sentido estricto. No es comunicación verbal, sino comunicación articulada. Esa articulación es la que permite que haya cosas y acciones sobre cosas —si se me permite la expresión—. La articulación comunicativa es bastante limitada, porque está determinada por la ejecución de la máquina. Absolutamente nada más. Sin embargo, tenemos que aprender a decir al ordenador qué es lo que tiene que hacer y con qué tiene que hacerlo.

C LA COMUNICACIÓN

Los ordenadores están por todas partes, en cada vez más cosas. Tendemos a que todo, en el fondo, esté controlado por un ordenador. No sé yo si será algo muy deseable. De hecho, tengo mis reservas. Los ordenadores que cada vez usa más la gente son los dispositivos móviles. De momento, los llamados teléfonos inteligentes y las tabletas[22]. Quizá en no mucho tiempo se generalizarán los relojes inteligentes. No me resisto

—aunque aquí no nos manchemos de tinta—, hay que saber manejar las cajas.

Entre sus capacidades destaca la mayor precisión en sus resultados. Este mismo texto —en su versión PDF— debería ser un ejemplo del mejor acabado al que acabo de aludir. Tomaré una muestra: la partición automática de palabras al final de línea. Por supuesto, eso es una propiedad que comparte con *Word*. Sin embargo, dicen que las apariencias engañan. *Word* parte las palabras según quepan en cada línea. ConTEXt pare las palabras según un cálculo global del párrafo completo. Es una calidad que no se ve, pero que se nota.

Un detalle que puede pasar desapercibido en la lectura. Antes de explicarlo, quede claro que un procesador de textos no es capaz de hacerlo. Toda la partición silábica de este texto en el documento PDF es automática. Aunque `pandoc` y todo lo que esté en tipos mecanográficos usa guión bajo, como podía hacerse con las máquinas de escribir en su momento. La existencia de un doble guión de separación automática de palabras sólo lo puede hacer un programa con funcionalidades de composición tipográfica. La razón de que lo use es que `pandoc` y `pan-doc` no son el mismo programa[4].

Además de la calidad de imprenta sus composiciones tipográficas, ConTEXt permite un tratamiento automatizado de textos extremadamente útil. Gracias a esta capacidad, es posible la generación automatizada de documentos. Con sólo pulsar una tecla, ConTEXt puede llegar a generar diferentes tipos de documentos de un mismo archivo de origen. O generar documentos mediante la combinación automatizada de datos. El método automatizado de edición digital no se basa en la combinación automatizada de datos[5]. Sin embargo, la edición digital automatizada es posible gracias que ConTEXt es un sistema de generación de documentos.

II EL EXPERIMENTO INICIAL

Después de la presentación de los personajes, es lógico que quien lea quiera ver datos. Porque de otro modo, parece que esto no es más es pura palabrería. Lo muestro con un ejemplo personal. Es el que me permitió ver la posibilidad de un método. Y que también era universalizable. Esto es, que podría usarlo más gente que yo.

Durante este último año, he tratado de encontrar un método de edición digital fiable. La búsqueda estaba motivada por la publicación en internet de mis propios escritos. Aunque tenía un interés específico en que el método fuese generalizable. Si me sirve a mí, ha de poder usarlo cualquier otra persona interesada. Al menos, cualquiera que escriba en alfabeto latino y textos no científicos.

A EL RESULTADO FINAL

Prefiero empezar por lo que he conseguido, que se puede resumir en la siguiente línea:

3 formatos × 2 versiones × 2 hojas = 4 minutos

Con un texto de unas 22500 palabras, se generan un total de ocho archivos automáticamente de un único archivo de origen. Hay tres formatos diferentes. Del texto existen dos versiones. De la versión para imprimir como libro, existen dos tamaños de hoja. Todo eso, pulsando una tecla y dejando que el ordenador trabaje. En total, ocho archivos en menos de cuatro minutos.

Explico los ocho archivos. Los formatos de archivo son ePub, MOBI y PDF. ePub es el formato habitual en lectores

algo mucho más sencillo de lo que parece. Se trata de aprender a usar un ordenador, en vez de entrenarse para usar un programa —o varios—. Ésa es la premisa básica que da título a este capítulo: «habla con la máquina». Quien prefiera, puede cambiarlo por «habla con el ordenador».

Evidentemente no me refiero a la charla de amistad íntima. No tendría sentido. Entre otras muchas cosas, porque no tenemos conocimientos suficientes para expresar intimidades en el código que desconocemos. Como nos pasaría con cualquier idioma extranjero. Por supuesto, esas posibilidades no están en el código informático. Se trata de poder comunicar órdenes, que no es sólo enunciarlas. Es necesario también saber responder a lo que la máquina nos comunique a su vez.

No pretendo que le demos un carácter personal al ordenador. Cuando empezamos a interpelar a la máquina —no sólo hablamos con nosotros mismos—, necesitamos un descanso. Porque la comunicación con el ordenador no es personal. Sin embargo, no deja de ser comunicación. Lo que tenemos que entender es qué no es personal en la comunicación que requiere la informática. Sólo así podremos manejar realmente ordenadores. De otro modo, nos situaremos ante una pantalla con teclado con la que tenemos práctica. Aunque sin embargo, estemos como delante del oráculo. Porque realmente ignoramos por dónde nos va a salir.

Existen muchas situaciones en que los ordenadores son exasperantes y desesperantes. En realidad, no se debe a la máquina misma, excepto en casos muy contados. La causa de esa profunda frustación es nuestra incapacidad de comunicación con la máquina. En realidad, un ordenador es algo tremendamente frío y calculador. Parece incluso contumaz en sus negativas. En el fondo, un ordenador es una grandísima

VIII HABLA CON ELLA

A UN NUEVO CONTEXTO

El ordenador es una máquina universal. En principio —afirma la teoría informática—, es capaz de cualquier operación no contradictoria. En Iberoamérica, a los ordenadores se les llama «computadoras». Son máquinas a nuestra entera disposición. Es muy probable que sean bastante competentes para tareas de oficina. Y según intento mostrar en estas páginas, son extraordinariamente competentes para tareas de edición digital.

El máximo rendimiento de esta máquina universal se consigue mediante la automatización de tareas. Como he tratado ya, eso supone pasar del uso de una herramienta al mandato a una máquina. Las órdenes son posibles no necesariamente porque se trate de máquinas inteligentes, sino porque son ejecutores altamente eficaces.

Como en toda comunicación, lo importante es que el mensaje —la orden, en este caso— la entienda la parte que tiene que actuar. El mensaje habrá de adaptarse a un lenguaje muy limitado, que propiamente es un código mucho más pobre que cualquier idioma. Con la articulación adecuada, es posible una comunicación altamente satisfactoria entre persona y máquina.

B LA MÁQUINA

Tras el párrafo anterior, quien lea puede mirar con recelo el texto. O pensar mal de su autor, dudando de su cordura. Es

de libros electrónicos. En realidad, es el estándar para libros electrónicos. MOBI es el formato que entiende *Amazon Kindle*. Y el formato PDF está pensado para impresión en papel. Para el documento PDF hay dos tamaños de papel: DIN A4 y carta. Las dos versiones son con notas y sin notas al final[6].

B ¿ES MUY LENTO?

La primera objeción al resultado anterior es que es un proceso muy lento. En parte, estoy de acuerdo. Aunque creo que la generación de todos los archivos no es lenta. Lo que es lento es mi ordenador. Es un ordenador portátil con diez años[7]. Entiendo que con un ordenador nuevo, los cuatro minutos se convertirán en menos de sesenta segundos. Por supuesto, con exactamente los mismos archivos.

La parte con la que no estoy de acuerdo es que el proceso en sí sea lento. Debemos ir paso a paso, porque hay diferentes formatos y diferentes programas en juego.

Los archivos ePub se generan en unos cinco segundos cada uno. Esa creación de archivos es muy rápida por una doble razón. El programa que los genera —`pandoc`— es extraordinariamente rápido. También la conversión no es en absoluto complicada.

La creación de cada uno de los archivos MOBI conlleva aproximadamente quince segundos. Es un proceso mucho más lento porque `pandoc` no convierte a ese formato, de momento[8]. *Calibre* es mucho más lento, porque trabaja de otra manera. Aunque la conversión no sea realmente muy distinta que la que hace `pandoc`.

La parte más lenta del proceso es la generación de los documentos PDF. Es un proceso mucho más complejo. No sólo porque aquí haya dos tamaños de hoja. Para generar una hoja

en la que haya un doble folleto, se necesitan tres pasos. Primero, se necesita el libro compuesto normalmente. Después, se tienen que ordenar todas las páginas del libro en un folleto. Finalmente, el folleto se repite en una página para que un folio —DIN A4 o carta— tenga dos cuartillas —DIN A5 o factura— con un folleto del texto cada una.

La necesidad de tres pasos provoca una generación más aparatosa para los documentos PDF. Además, el mayor tiempo se debe a que PDF es un formato fijo. Sólo generar el documento PDF del primer paso lleva unos treinta segundos. PDF es un formato que necesita que todo esté terminado. El documento PDF tiene que tener todo determinado cuando se genera[9]. Por contraste, ePub sólo tiene el texto, que se determina en el momento de su lectura, no cuando se crea. Por eso, todos los cálculos necesarios para crear el documento PDF redundan en mayor tiempo de proceso.

Como ya he dicho, generar el primer documento PDF conlleva unos treinta segundos. Componer un folleto de ese primer documento supone unos cinco segundos. Crear un doble folleto de ese segundo documento PDF tarda un máximo diez segundos. En total, generar cada documento PDF final —contando los tres pasos— supone cuarenta y cinco segundos cada uno.

Hagamos las cuentas por partes, del proceso más lento al más rápido. Cuarenta y cinco segundos por cuatro documentos PDF —con y sin notas, ambos en tamaños DIN A4 y carta— dan tres minutos. Quince segundos de cada uno de las dos archivos MOBI —con y sin notas— suman medio minuto. Y cinco segundos de cada uno de los dos archivos ePub —también con y sin notas— suponen diez segundos. En total son tres minutos y cuarenta segundos[10].

elemento individual se le aplicará la configuración genérica. El formato por grupos depende de que pertenezca a una u otra clase determinada. Y tendrá formato singular, si tiene el identificador único que corresponda.

Las propiedades son características de cada formato. Porque no todos los formatos disponen de los elementos de igual manera. Por ejemplo, los saltos o el tamaño de página tienen sentido en formatos que puedan usarlos. O existen propiedades que están disponibles en un determinado formato y no en otros. El ejemplo sería el distinto tipo de guión para la partición silábica al que me he referido en *I.B.3. ConTEXt*.

Con las propiedades y sus valores comenzamos damos formato al texto digital. El ejemplo de edición electrónica de estas páginas usa dos formatos: ePub y PDF. El primer formato es estándar en los lectores de libros electrónicos. El segundo formato está pensado para la impresión en papel. El formato no puede ser común a ambos tipos de archivo. La apariencia podrá ser común, pero cada tipo de archivo ha que tener su especificación por separado. El formato de cada tipo de archivo contendrá las instrucciones propias de cómo tratar cada elemento.

texto. Tener un número limitado de elementos, con posibilidad de atributos, es lo que evita tener que reinventar la rueda con cada edición digital.

D PROPIEDADES

Las propiedades de un elemento son las distintas posibilidades de formato de un elemento. Con un ejemplo sencillo, las propiedades de un título —de cualquier nivel— pueden ser: tipo de letra, alineación del párrafo del título, espacio anterior, espacio posterior, márgenes izquierdo y derecho. Cada propiedad tiene un conjunto de valores posibles. Izquierda, derecha, centro o justificado son valores de alineación de párrafo. Los elementos no comparten las mismas propiedades. Tampoco los valores son comunes a todos los elementos.

El formato de los elementos puede ser genérico, por grupos o individual. Formato genérico es aquél en que todos los elementos tienen un único y el mismo formato. Un ejemplo sería todas las expresiones enfatizadas en cursiva. Formato por grupos sería en que los elementos de una clase comparten un mismo formato. Un ejemplo de clase serían el grupo de los títulos sin numeración. Formato individual es el que es propio del elemento que tiene un identificador único. Por tanto, ese formato no lo comparte —ni lo puede compartir— con ningún otro caso del mismo elemento. El colofón de un libro estará centrado y tendrá un ancho menor, ya que es el único elemento al que se le debe aplicar esa configuración en todo el documento.

Las propiedades que cada elemento individual tenga, depende de la configuración general. Los tres tipos de formato —genérico, por grupos e individual— no son incompatibles. Pueden darse las tres en un mismo documento, aunque no se apliquen las tres completamente a todos los elementos. A todo

C UN PROCESO AUTOMÁTICO

La ventaja de este modo de proceder es poder trabajar en un solo origen todos los documentos que necesitemos. Por eso, en un momento dado, podemos pedir que se generen todos los archivos. Aunque lo habitual no será que necesitemos todos los archivos.

Mientras vamos redactando el texto, es fácil que sólo necesitemos un archivo de salida y no todos. Para ir comprobando cómo queda el texto final. La automatización es precisamente que usamos un solo archivo y podemos obtener directamente el archivo de destino que queramos. Por limitarnos a los formatos que he mencionado antes, puede ser el documento PDF o el archivo ePub. Con la presentación final que va a tener cada uno de los archivos. Sólo está a una pulsación de la tecla que elijamos.

Este es un método de trabajo que permite resultados de alta calidad. Si alguien lo prefiere así: con resultados profesionales. Estas páginas tratarán de mostrar cómo de un único origen se pueden conseguir varios archivos en diferentes formatos.

Lo único que es necesario son las instrucciones de configuración de cada uno de los tipos de archivo. Así, cada formato tendrá sus instrucciones propias y el diseño será el más adecuado. Cada formato tendrá exclusivamente los datos necesarios. Sin que sobre o falte nada.

III EN DONDE QUIERAS

Como suele pasar en estos casos, dejé a una persona el primer borrador de algo que quería publicar en internet. Como somos buenos amigos, me dijo que lo leería con gusto en cuanto pudiese. En la misma conversación, tras hojearlo rápidamente, me preguntó con qué programa lo había hecho. Le contesté que con `pandoc`. Me dijo que no le sonaba y me preguntó si era algo de *Linux*. Le dije que no, que se podía usar con cualquier sistema operativo.

La gran ventaja del método del que estoy hablando es que puede usarse con cualquier sistema operativo. Al menos, seguro con *Windows*, *MacOS X* o *Linux*. Habría otros sistemas operativos, pero si los usas, seguro que sabes cómo hacer que los programas funcionen en esos sistemas.

Personalmente me gusta mucho más *Linux* que cualquier otro sistema operativo. *MacOS X* me parece que vende algo que está más allá de la informática misma. Yo a un ordenador le pido funcionalidad. El diseño para mí no es absolutamente prioritario. Especialmente si tengo que pagar por él. En cuanto a *Windows*, me temo que lo que he oído de sus últimas versiones no me incita a usarlo. Incluso aunque quisiese usar la última versión, primero tendría que cambiar de ordenador. Con el que tengo, puedo usar la última versión de *Linux*[11]. Con *MacOS X* o *Windows* es sencillamente implanteable.

No pretendo convencer a nadie de que use *Linux*. Yo creo que es una opción mucho mejor. Sin embargo, cada cual ha de tomar sus propias decisiones cuando alcanza la edad adulta. Además, *Linux* tendría sentido para cualquier otra cosa. No

saber a dónde ir. Algo tan sencillo como una referencia cruzada, necesita que el elemento disponga de un tipo de atributo llamado «identificador único»[21].

Es importante poder distinguir también un grupo del resto de casos de un mismo elemento. Como ya he dicho, puede que queramos que un grupo de títulos no aparezca numerado. O que en ciertos capítulos, aunque haya un título, que no aparezca únicamente en los marcadores. Los documentos PDF y ePub tienen marcadores que pueden ser útiles para mostrar títulos páginas como la de derechos de autor, cita, dedicatoria y colofón. Sin embargo, carecería absolutamente de sentido incluir esos títulos en el texto o en el índice general. ¿Se trata de títulos de capítulos en los dos casos? Por supuesto, se trata del mismo elemento, pero es esencial poder establecer clases dentro del mismo elemento.

La finalidad de las clases es distinguir entre distintos tipos del mismo elemento. No es la individualización del identificador único —aunque nada impide que la clase de un elemento no tenga ningún caso o sólo uno—. Sin las clases, habría que disponer de muchos más elementos. Trabajar con más elementos es mucho más engorroso. Es mejor tener clases dentro de un mismo elemento. Eso permite que que todas las clases del mismo elemento sean sólo uno, en parte iguales y en parte distintas.

De este modo, trabajaremos con un número muy determinado de elementos. Y en los distintos elementos se articulará con atributos. Las clases son necesarias en los elementos para evitar tener que crear una infinidad de elementos distintos. Si los diferentes elementos son demasiados y no están bien articulados entre sí, la estructura lógica del documento se resentirá. Y será mucho más engorroso editar digitalmente un

ser humano podría hacerlo. ¿O cómo distinguiríamos títulos, por ejemplo, si todos los párrafos fuesen exactamente iguales?

En definitiva, un texto completo ha de incluir elementos. El texto digital también tiene que tenerlos. De este modo, parte del proceso de redacción es distinguir los diferentes elementos. Es necesario especificar qué es cada cosa en el texto. Después determinaremos cómo ha de aparecer. Es necesario separar la redacción del texto de su presentación final. El texto no es su formato, pero sí sus elementos. De esta forma, se distinguen dos momentos: la escritura y la composición tipográfica. Primero, escribimos el texto y después le damos forma. Son dos operaciones distintas, aunque puedan llegar a desarrollarse simultáneamente.

C ATRIBUTOS

Si sólo tuviésemos elementos sin más, su número sería potencialmente infinito. Normalmente, un elemento aparece varias veces en un mismo documento. Un elemento tiene múltiples casos. Un ejemplo sería el título de capítulo —o de primer nivel—, lo habitual es que aparezca varias veces en un libro. Sin atributos, sería imposible referirnos a un capítulo concreto o que exista una serie de títulos de capítulos sin numeración.

Quien lea puede preguntarse qué relevancia tiene referirnos a un elemento concreto. O cabe interrogarse para qué sirve distinguir un grupo del resto de casos del mismo elemento. Se trata de dos posibilidades básicas e irreductibles entre sí. La referencia a un elemento singular es necesaria para poder crear un enlace a ese elemento. El vínculo sólo podrá funcionar, si el elemento tiene un identificador único. Si la identificación no es única, el destino del enlace es irresoluble. Si hay varios destinos con el mismo nombre —o ninguno—, es imposible

hace en absoluto falta para el método automatizado de edición electrónica que describo.

La perspectiva es que las herramientas necesarias para trabajar edición digital se desarrollan para cualquier sistema operativo. En el que te encuentres más a gusto, allí sólo tendrás que instalarlas y comenzar a disfrutarlas.

Eso tiene otra gran ventaja, que los archivos son universales. Esto es, puedo empezar trabajando en *Windows*, pasar a *MacOS X* y luego terminar en *Linux*. En períodos largos, o varias veces al día. Ahí está la gran potencialidad de trabajar con estándares. Por supuesto, en cualquier sistema tenemos que tener los archivos necesarios para el trabajo y los programas adecuados a nuestro sistema operativo[12]. Si los tenemos, da igual el sistema operativo que usemos.

Eso permite que un mismo grupo de trabajo pueda tener ordenadores con distintos sistemas operativos. Y que en todos pueda trabajarse lo mismo. Eso es compatibilidad real.

IV ¿POR QUÉ NO *WORD*?

Hace unos meses, una persona me envió un documento PDF bastante extenso. Mi intención era imprimirlo para poder leerlo y corregir las erratas que me pudiese encontrar. Antes de imprimirlo, me fijé en algo muy extraño. El tamaño de hoja que tenía el documento era carta. Es un formato americano —me refiero al continente, no a sus Estados Unidos—. En Europa usamos el tamaño de papel DIN A4. Son tamaños parecidos, pero no iguales[13]. Sobre todo, no son proporcionales. La carta americana es ligeramente más ancha y más corta que el DIN A4 europeo.

Como me interesaba que la impresión saliese bien, me puse en contacto con la persona. Le pedí si me podía enviar el archivo *Word* con el que había generado el docuento PDF. Le expliqué que el tamaño de hoja en el documento PDF era incorrecto. Para mi sorpresa, me dijo que no podía hacer nada porque los índices los había hecho a mano. Si cambiaba el tamaño de página, cambiaría la numeración de las páginas y habría que rehacer los índices. Era consciente de que el tamaño de página estaba mal, aunque no sabía por qué.

Además del error, todavía no salgo de mi asombro por algo muy básico. No son los ordenadores, son las personas. Un ordenador es una máquina de automatizar tareas. Si algo está claro es que un índice es algo que debe hacer un ordenador. No lo digo en el caso concreto de esa persona, sino en general. Todo lo que un ordenador pueda hacer es mejor que no lo hagamos las personas. Será más rápido y tendrá menos fallos. O si no podemos preguntarnos cuántos fallos

hay. Incluyendo elementos en el texto, lo que indicamos al ordenador es qué hay, no necesariamente cómo ha de manejarlo. O más propiamente, determinando qué hay, podremos especificar después cómo habrá de tratarlo el ordenador.

Elemento es cada una de las partes que estructuran lógicamente un texto. Por supuesto, podríamos hablar de objetos. Aunque en informática es un término que debe evitarse, porque «objeto» tiene una connotación muy precisa. La estructura lógica a la que me refiero es a la que comparten personas y ordenadores. Son elementos: los títulos de diferentes niveles, las secciones de texto en sus diferentes niveles —partes, capítulos, secciones…—, los diferentes tipos de notas —al pie, al final, e incluso a a la línea—, la página, los párrafos… Por supuesto, la lista no es completa. Sin embargo, hace referencia a una gran variedad de elementos posibles. Cada elemento nos permite especificar en el texto —hablando mal y pronto— qué es cada cosa.

Los elementos están en el texto. En el archivo informático hay que marcarlos de modo explícito. Así, quien escribe sabe que el ordenador podrá articular cada uno de los elementos en la edición digital del texto. De otro modo —repito con insistencia—, no habría nada que articular, que presentar de un modo diferente. Para poder distinguir, es necesario marcar qué es cada cosa. Así lo sabe el ordenador y quien escribe.

Los elementos no son un añadido al texto digital. Son parte del texto mismo. En el caso de la edición digital, han de ser parte del texto informático. De otro modo, es imposible que un ordenador pueda procesarlos. Especificar qué es cada cosa es parte integrante de la actividad de redacción. De otro modo, no es que un ordenador no pueda distinguir. Ningún

una nota al pie o una nota al final. Por supuesto, sólo por el formato. Creo que sería bastante complicado. Desde luego, no lo voy a intentar aquí. El enfoque visual se muestra mucho más complicado que el nombre. Qué es el elemento es mucho más directo que tratar de ver cómo está.

B ELEMENTOS

Todo texto tiene elementos. Al menos, palabras y párrafos. Un texto, manuscrito y mecanografiado en ordenador, contiene como mínimo ese par de elementos comunes. ¿Cómo los distinguimos? Las palabras se separan con espacios, aunque pueda haber otros signos de puntuación en medio. Los párrafos se distinguen por los saltos de línea intermedios. Los podemos reconocer y el ordenador puede distinguir palabras y párrafos porque los marcamos. Exactamente, con cada espacio o con cada salto de línea. Ambas distinciones son esenciales para componer automáticamente cualquier tipo de párrafo. Además de que sirvan para que las personas los reconozcamos al escribir o leer el texto.

Como palabras y párrafos se distinguen igual en la escritura a mano o en ordenador, no nos resulta difícil reconocer esos elementos. La estructura lógica de un texto necesita muchos más elementos. En la escritura a mano, muchos de estos elementos sencillamente no existen. No escribimos con cursivas, ni tampoco añadimos notas al pie o al final en un manuscrito. Otros elementos se articulan de diferente manera en un texto manuscrito que en un archivo informático.

La flexibilidad y maleabilidad de un texto digital depende de incluir correctamente cada uno de los elementos necesarios. El principio básico es que lo que no se distingue el archivo de origen, no se podrá presentar después como diverso. Diciéndolo de modo muy simple: no se puede sacar lo que no

puede haber en un índice con aproximadamente doscientas secciones. Un ordenador lo hace muchísimo más rápido y sin errores. Como añadido, pone enlaces si generamos un documento PDF medianamiente bien hecho. No se trata de saber cómo conseguirlo en un programa concreto. Porque eso se busca en internet, si hace falta. Lo relevante es saber qué es lo que tenemos que encargar al el ordenador que haga.

Volviendo al error del programa, como no uso *Word* habitualmente, tardé tiempo en poder comprobar que el fallo está en el mismo *Word*. Al parecer, los programas de *Microsoft Office 2010* determinan que el tamaño de página por defecto sea carta en vez de DIN A4. En Europa, el error es grave. Si bien es un detalle, ni *Microsoft* es una empresa novata, ni *Office* ha finalizado hace un mes su primera versión de pruebas. Es un error de programación, que supongo que habrán corregido. Si bien, no lo sé y me afecta sólo indirectamente.

A EL PROCESADOR DE TEXTOS

La limitación de *Word* no es que la gente de *Microsoft* haya tenido patinazo. Porque se podrían excusar alegando que la mayoría de quienes lo usan ni se han dado cuenta de ese fallo. En mi caso, la gente a la que le he preguntado para ver el fallo no sabía muy bien de qué les estaba hablando. En el fondo, siempre nos podemos excusar diciendo que los tamaños son más o menos iguales.

El detalle del tamaño de papel incorrecto puede ser irrelevante en la mayoría de los casos. Porque la gente, aunque note algo extraño, no sabrá qué es. Si hablamos de edición digital, pasar por alto ese detalle es sencillamente una chapuza. No da lo mismo, porque es importante acabar bien el trabajo. La edición digital exige cuidar los detalles.

Microsoft Word es tan poco adecuado para edición digital como lo es *LibreOffice Writer*. No es una cuestión de marcas. Tampoco es cuestión de licencias de pago o gratuitas. Ni siquiera se trata del modelo de desarrollo informático abierto o cerrado. El procesador de textos como programa es lo que falla. Porque un procesador de textos se puede usar para tomar notas. En el fondo, se puede usar como si fuese una máquina de escribir con memoria. Pero si hay que editar digitalmente textos, es muy fácil que el procesador de textos se quede corto como herramienta.

La referencia a la máquina de escribir puede sonar casi a la época de los dinosaurios. La limitación a la que me refiero es sencillamente que escribir no es lo mismo que editar. Editar digitalmente supone componer tipográficamente un texto. No es sólo que quede bonito el texto que hemos —o han— mecanografiado. Hay que organizar visualmente el texto. No es redacción, puesto que el texto está completo. Tampoco es diseño, porque se trata de legibilidad.

La imagen del procesador de textos mezcla cuestiones distintas: texto y presentación. Son distintas, porque son actividades distintas. El texto hay que escribirlo. Si lo escribimos en un ordenador, lo mecanografiamos. Luego debemos organizar su presentación para lectura. La composición tipográfica del texto es distinta de su redacción. Con un procesador de textos, esas dos fases o actividades parecen más difícilmente distinguibles.

La composición tipográfica que requiere la edición digital necesita que el texto y sus elementos estén dados. Para poder organizar visualmente un texto tenemos que saber qué es cada elemento. No sólo palabras y párrafos, sino títulos, notas al pie, referencias cruzadas y más cosas. Para que luego digamos al

es mucho más fácil para la persona y la máquina hacerlos más maleables. Si sólo tenemos descripciones de elementos, estarán fijados y será más difícil que podamos cambiarlos. Tanto dentro del mismo formato de archivo, como si tratamos de convertirlos a otros tipos de archivo.

Definir los elementos de un texto por su formato es como pintarlos. Cada vez que sólo decimos cómo es y no qué es, se le añade una capa de pintura. Pero cada capa de pintura nos esconder realmente la cosa. Y hay un momento en que tenemos que rascar todas esas capas de pintura. Porque para trabajar con textos digitales son escasamente útiles. Con el ejemplo del título: un texto centrado y con cursiva no identifica a un título. Porque no será la única clase de texto en cursiva o que esté centrado. Ningún elemento de un texto se agota en su apariencia. Cada brochazo de pintura nos deja el elemento lógicamente menos reconocible.

Si un ordenador es un asistente eficacísimo y escasamente inteligente, la mejor manera es nombrarle cada cosa, no describírsela. Porque no es capaz de distinguir. Pongamos otro ejemplo, como la cursiva. Realmente, la cursiva es un formato, no un elemento. En cursiva podemos enfatizar o marcar un pasaje en idioma extranjero. O puede ser el formato —o parte del formato— de un título. Si decimos cursiva, el ordenador no sabrá a qué pertence: si al énfasis, a un pasaje en idioma extranjero o al título. Es mucho más fácil nombrar cada elemento. Luego ya se determinará el formato. Y podrá cambiarse de un modo mucho más fácil.

La importancia de estructurar el texto en elementos y no por formatos se puede ver en un ejemplo. Creo que muestra con la máxima claridad que el formato es un segundo momento a la estructuración del texto en elementos. Intenta describir

La diferencia entre el qué y el cómo en un texto se puede apreciar en la diversidad entre nombrar y describir. Hago una aclaración importante para evitar malentenidos. Estamos tratando de edición digital de textos. Aunque el vocabulario pueda ser común, no pretendo hacer teoría literaria o hermenéutica. Sólo trato de articular la estructura presente en el texto para su elaboración informática.

Cuando vemos un texto, observamos que está compuesto de partes. Al menos, de letras, palabras, líneas y párrafos. Hasta ese nivel puede llegar una máquina de escribir. Un texto editado puede tener títulos, encabezados, números de página, notas al pie, pasajes enfatizados y expresiones en lenguas extranjeras. Por supuesto, la lista anterior no es completa.

Ahora vamos a ver la diferencia entre el qué y el cómo con un par de ejemplos. Tomemos un título de capítulo. Podemos decir que es un título de primer nivel. O de segundo, si es que hay títulos para las partes del texto. Eso es nombrar y saber qué es el elemento al que nos referimos. También podemos decir cómo está: centrado, en negrita, cursiva o versalitas, con un tipo de tamaño superior. El título tiene un formato especial para que pueda reconocerse en la lectura. Sin embargo, el título no se agota en su formato. Aunque sólo sea porque un título de capítulo es fácil que tenga diferentes formatos en diferentes textos.

Si decimos qué es, luego podremos determinar cuál ha de ser su apariencia. Propiamente, ordenaremos su formato. Si sólo sabemos cómo es, es mucho más difícil trabajar con el texto. Aunque ni siquiera haya que cambiar el formato. En ese caso, el texto sólo puede tener una salida. Ése es el esquema de un procesador de texto: sus documentos sólo pueden imprimirse[20]. Si el texto tiene elementos con su nombre propio, programa cómo ha de tratar cada elemento. Sobre este punto volveremos más adelante.

El procesador de textos se queda corto por la confusión. No permite distinguir entre el texto, sus elementos y su formato. Así es muy difícil automatizar tareas. Porque si todo es visual y deíctico —que señala apuntando, con el dedo o con el puntero—, no hay manera de expresar una orden general. En el fondo, los programas visuales —con ventanas— sólo permiten decir: «esto». El ratón es un puntero y por eso sólo permite señalar y marcar. Voy a ilustrar el par de frases anteriores con el experimento que relato en este capítulo. Aunque lo voy a contar de una manera diferente. No desde el resultado, desde el proceso.

Para poder tener qué editar, tuve que escribir el texto. En el mecanografiado, tengo cuidado en distinguir todo lo que luego quiero que sea distinto. No sólo teclear correctamente el texto, sino marcar los títulos, notas al pie, énfasis, direcciones de internet, capítulos especiales… así todo. Lo indistinguible no puede aparecer como distinto en el archivo final. Luego, especifiqué la apariencia que tendría que tener cada elemento en el documento PDF y en el archivo ePub. No es configurar cada elemento individual. Es ordenar, por ejemplo, que los títulos de capítulo vayan centrados. Así con todos los elementos.

Para terminar, escribí las órdenes para que se crease cada uno de los ocho archivos finales. Aunque el archivo de origen era único, cada tipo de configuración tiene que ir con su clase de documento. Guardé todas esas órdenes para poder ejecutarlo siempre que quisiera. Así, en menos de un segundo, tengo todo automatizado y se genera a la máxima velocidad que da el ordenador.

Con un procesador de textos, la historia hubiese sido ligeramente diferente. Tendría que mecanografiar el texto. Para formatearlo habría que ir seleccionando con el ratón e ir seleccionando elementos o formato. Seleccionar elementos es decir qué es el texto seleccionado: un título, una nota al pie, una dirección de internet o lo que sea. Seleccionar el formato es decir cómo va a estar el texto seleccionado. Lo peligroso de seleccionar directamente el formato es que si hay que cambiar algo, nos veremos en la obligación de cambiar cada uno de los casos en donde hemos aplicado el formato.

Hasta aquí un procesador de texto no es tan distinto. El problema es que no puedo tener configuraciones distintas para un mismo texto en un mismo archivo. Aquí comienzan los problemas. Porque tendríamos que tener dos tipos de archivo. Aunque en mi caso necesitaba ocho archivos diferentes de salida. Me temo que tendría que tener al menos cuatro para luego poder generar los ocho. Aquí empiezan los dolores de cabeza. Porque si hay que cambiar algo, aunque sea corregir una errata, tenemos un problema. Una corrección multiplicada por el número de archivos existentes del mismo texto.

Supongamos que me las hubiese arreglado con cuatro archivos. Generar los ocho archivos es otro problema. Porque hay que abrir uno por uno, seleccionar la opción correspondiente y grabar. Por muy rápido que fuese, es imposible que eso supere a órdenes escritas grabadas. Porque sin intervención humana, el ordenador va a la máxima velocidad. Por definición, nadie es más rápido que la velocidad máxima. Y en la práctica, nadie es tan rápido como una máquina. Todo eso, considerando que la conversión de archivos funcione perfectamente.

VII QUÉ ES, NO CÓMO ESTÁ

Toda la dificultad de la edición digital de textos está en un principio básico. Este principio nos permite dejar atrás la perspectiva única del procesador de textos. Se trata de un principio universal de la comprensión. Lo enuncia Platón: la cuestión no es cómo son las cosas, sino qué son[19]. En esto se basa todo el trabajo de edición digital. Sabiendo qué es cada cosa, luego pensaremos cómo hemos de presentarla.

En este capítulo está la clave de la edición digital de textos. En realidad, es algo asombrosamente sencillo. Precisamente por su simplicidad, es una característica que posee también una gran profundidad. Creo que la explicación es muy clara. Una vez hayamos comprendido esta parte, nos será mucho más fácil editar digitalmente textos. Se trata de ver la sencillez de la perspectiva de cómo está articulado un texto. Esa estructura transparente permite trabajemos del mejor modo posible un texto con un ordenador.

A UNA BUENA REFERENCIA

Por si hubiese alguna reticencia en la lectura, no pretendo en absoluto hablar de filosofía. Si hago referencia a Platón, es exclusivamente por la advertencia que nos lanza de no confundir el qué con el cómo. Se trata de un pensamiento profundo. Contra lo que demasiadas veces pensamos, profundo no es sinónimo de oscuro. Profundo es lo que llega al fondo de las cosas. Tomadas tal cual son, las cosas son sencillas. De ahí que la intuición platónica sea la propia de una persona sabia.

realmente lo que tienes. Has de deducirlo de lo que ves en pantalla.

D RECURSOS

Una limitación de un procesador de textos es la necesidad de mayores recursos que un enfoque puramente lógico. La razón de fondo es muy sencilla: hacer algo es más lento que no hacerlo. Pedirle al ordenador que dibuje la ventana de un programa es más lento que no pedírselo. No está mal usar los recursos. No es buena idea usarlos mal.

Cuando abrimos el procesador de textos, el programa tiene que cargar una serie de recursos para poder usarlos. Pondré un ejemplo, la partición silábica al final de la línea. Debe ser automática en todos los casos. Un procesador la tiene que tener disponible siempre. O la corrección ortográfica automática siempre tiene que estar trabajando. Así el ordenador necesita más recursos para trabajar. Porque realmente tiene que tener disponible lo que pueda usar. No usa sólo lo que tiene que ejecutar sin más.

La edición digital de textos trabaja de un modo ligeramente distinto. Por un lado escribiríamos el texto sin más. Y pediríamos al programa adecuado que genere el archivo adecuado. De un texto con órdenes, pasamos a tener un texto compuesto tipográficamente. En el momento de la generación, el programa procesa el archivo de origen y genera el documento necesario. La diferencia estriba en que un procesador lo mezcla todo. Tanto el texto y las órdenes en la imagen final de la página, como la redacción del texto y su composición tipográfica. Esa mezcla impone al final muchas limitaciones.

B EL PROGRAMA

Los procesadores de textos son programas ingeniosos. Porque permiten un uso pasable sin especiales conocimientos informáticos. Sin embargo, del mismo modo que facilitan la tarea inicial, limitan su alcance o su funcionalidad. Si existiese el alemán sin esfuerzo —como prometía el método *Assimil*—, literalmente todo el mundo hablaría alemán. Con los procesadores de texto pasa algo parecido. Sin embargo, esfuerzo y sufrimiento no tienen que ser sinónimos. Lo que un procesador de textos nos muestra como muy fácil también esconde otra parte que nos impide sacar más rendimiento al programa.

Toda la informática se compone esencialmente de dos elementos: órdenes y datos. En un texto informatizado, por tanto, estos elementos serían: texto y órdenes de configuración. Un procesador de textos nos muestra el texto con el resultado de las órdenes de formato ya aplicadas. Eso supone una limitación. Porque no es extraño que en alguna situación veamos que algo falla, pero no sepamos por qué. No pocas veces en esos casos, lo que pasa es que hay una orden de configuración que no advertimos que está aplicada.

Esa ocultación que hace el procesador de textos delimita hasta dónde puede llegar quien lo use. Esas limitaciones afectan de modo esencial a la edición digital y a la automatización. Porque además, las limitaciones están también presentes en el sistema de entrada y la conversión a otros formatos.

1 EL FORMATO DE ENTRADA

Un documento de *Microsoft Word* —o *LibreOffice Writer*— contiene texto. Sólo texto, estaríamos tentados a decir. Lo lógico es que fuese así. La realidad es quizá un poco distinta. Y aquí las imágenes no son el problema. De hecho, un archivo de

Word es un archivo comprimido. En su interior contiene más archivos. Por curiosidad, podemos ver qué tienen dentro[14]. Sin embargo, esos archivos son inútiles para usarlos directamente. Porque si cambiamos algo e introducimos un error, podemos corromper el archivo.

Ver sólo el texto con las ordenes ya aplicadas nos puede dar más problemas. Porque el texto completo es el escrito más las órdenes. En muchos casos, puede haber una interacción que no entendamos. Y algo que es muy fácil con texto puro, como un historial de cambios, es engorroso con el archivo de un procesador de textos. En el fondo, no trabajamos directamente con el texto, sino con la representación que nos da el procesador de textos. Sí, tiene sus ventajas en algún caso. Sin embargo, impone un peaje a partir de un cierto nivel.

2 los formatos de salida

Mi experiencia me dice que con procesadores de texto es importante siempre usar el formato propio. No sólo pasa con procesadores de texto. Sin embargo, con procesadores de texto es especialmente importante. Porque añaden muchas órdenes irrelevantes. De otro modo, haríamos una doble conversión cada vez que lo cambiásemos: al abrirlo y al guardarlo.

Los procesadores de textos tienen un formato nativo para guardar textos. En el fondo, ese formato está pensado para almacenar e intercambiar textos. Empezar a convertir puede ser empezar a probar. Porque el grado de precisión que teníamos en el formato anterior puede empezar a fallar.

Probablemente la mejor conversión desde un procesador de textos es a un documento PDF. Porque en realidad, esa conversión lo que hace es imprimir todo a un archivo. Lo que ahí se necesita es un buen generador de documentos PDF.

La inmediatez de la que carece un entorno visual está en formular una orden. Dicho mal y pronto, en pedirle al ordenador que haga algo. La petición es más lenta, porque es menos ágil. También es más lento y es más opaca. La imagen del texto que nos da un procesador de textos es el resultado final. Como en el siguiente caso:

> Esto es *cursiva* y esto es `código`.

El párrafo anterior contiene las dos órdenes —cursiva y mecanografiado— ya aplicadas. Es más claro en el resultado, que las órdenes en un entorno lógico —donde hay que escribirlas—, como en el párrafo siguiente:

```
Esto es _cursiva_ y esto es `código`.
```

El código anterior es más claro. En realidad, es transparente. Porque se sabe exactamente donde empieza y donde termina la cursiva y el mecanografiado. Si parece un perfeccionismo excesivo, no conviene olvidar que se trata de edición digital. Si el punto final estuviese con tipo mecanografiado es muy difícil reconocerlo a simple vista. Lo mismo pasaría si después de la palabra en cursiva hubiese una coma. Además, hay un error muy común con un procesador de textos. Si tenemos que añadir texto justo después de la palabra en cursiva o la mecanografiada, es fácil que sigan con esos formatos. Si hay caracteres que muestran el principio y el fin de una orden, está claro dónde hemos de poner el cursor y teclear.

En el caso de textos, la imagen visual es parecida a la del papel en el carro de una máquina de escribir. La diferencia en es que en una máquina de escribir ves todo lo que tienes. En un procesador de textos la mediación es opaca. Te muestra el resultado de todo lo que tienes. Sin embargo no te muestra

para complacer a quien usa el ordenador. La capacidad de mandato está mucho más limitada. Igual que en la relación con las personas, podemos decir menos si señalamos cosas y movemos los brazos, que si hablamos. Al necesitar una traducción a imágenes, el entorno de escritorio toma un rodeo. Por tanto, no puede ser sino más lento que la orden directa. Si se me permite la ironía, es como si pidiésemos al ordenador cosas trantando de evitar el mandato directo.

Con rodeos, sólo se retrasa la persona, no el ordenador. Se retrasa la comunicación de la orden que ha de ejecutarse. Esa mediación visual puede ser un límite antes de tiempo. No lo pone la máquina, se lo ponemos a la persona. Porque una interacción más fácil es también una interacción más lenta. Porque teclear normalmente es mucho más rápido.

Por supuesto, habría muchos ejemplos. Escogeré uno que muestra bien la diferencia, las notas al pie o al final. En un procesador de textos hay que pinchar en el menú Insertar, seleccionar Nota al pie / Nota al final. Nos aparecerá una ventana de diálogo y debemos pulsar Intro. Después de escribir la nota, debemos colocar el cursor en el cuerpo del texto. Con un sistema lógico, una nota al pie necesitaría sólo los tres caracteres siguientes —el texto de la nota iría entre los corchetes—:

```
^[]
```

Mi pregunta es simple: ¿alguien cree que la segunda opción es más lenta o más engorrosa que la primera? Por supuesto, necesita un período de adaptación para usarlo con agilidad. Con una nota, la diferencia es inapreciable. En un informe o una tesis doctoral, quizá el número de notas puede ser alto. Es muy fácil superar el centenar, doblarlo o incluso triplicarlo. Ahí, la diferencia en la facilidad y rapidez de uso es muy relevante.

Pero eso es externo al programa. Aunque haya programas que lo incorporen, como *LibreOffice* o las últimas versiones de *Microsoft Office*.

El contraste claro es la conversión a formatos que no sean de procesadores de textos. Por ejemplo, al formato de libros electrónicos: ePub. En ese caso, es muy fácil que los procesadores añadan información multiplicada. Es innecesaria porque es redundante, aunque el resultado no sea erróneo en sentido estricto. Sin embargo, con más información, enlentece su procesamiento por parte del ordenador. Porque tiene más órdenes que leer y que cumplir. Quizá no se note tanto en un ordenador, pero puede marcar una gran diferencia en un lector de libros electrónicos.

C LO MEJOR DE CADA CASA

Dicen que una buena filosofía de programación es que cada programa haga una única cosa y que la haga lo mejor posible. Muy probablemente ésa es la formulación en positivo de los límites del procesador de textos. Se trata de usar la mejor herramienta disponible para cada tarea. Si estamos hablando de un método automatizado de edición digital, es probable que haya que usar más de un programa.

La edición supone hacer disponible un texto para todo el mundo. Con la imprenta, era mediante el papel. Mediante internet, se trata de hacerlo en aquellos formatos de archivo que permitan tanto la impresión de calidad, como la lectura en dispositivos especiales. Por eso, editar digitalmente es poder editar en formatos PDF, ePub y MOBI.

Mi experiencia me lleva a escoger los tres mejores programas para esa triple tarea. Como ya he dicho, son ConTEXt,

`pandoc` y *Calibre*. Son tres programas independientes, pero puede trabajarse con ellos de modo coordinado.

`pandoc` trabaja el archivo origen del texto y genera el formato ePub. Puede generar otros muchos formatos. De momento, no puede generar formato MOBI, por eso es necesario usar*Calibre*. Y del mismo origen, `pandoc` puede generar un archivo que ConTEXt compila en un documento PDF de alta calidad tipográfica.

diez archivos, es una tarea ligeramente molesta. Sin son cien archivos, empieza a ser un problema grande. Sin son mil archivos, es imposible. Con la orden adecuada, nos puede llevar segundos hacerlo. O aunque nos lleve minutos, se los llevará al ordenador, no a quien lo manejamos. Dejamos trabajar al ordenador, que cuando acabe, acabó.

El ordenador no es un asistente propiamente inteligente. Sin embargo, es un grandísimo ejecutor de tareas. Con imágenes, señalando con el ratón, la posibilidad de comunicación es mucho más limitada. La capacidad de interacción es mucho mayor si hay comunicación por palabras. Al menos hay verbos y nombres de objetos. Eso es una nueva dimensión. Pongo otro ejemplo. Imaginemos que queramos abrir todos los documentos PDF modificados hoy. Con un ratón hay que buscar los archivos PDF, ver los que están modificados hoy y abrirlos. Ninguna persona es más rápida que un ordenador en buscar ni los archivos PDF, ni los que han sido modificados hoy.

Manejar un ordenador es hacer que trabaje para nosotros, así nos centrarnos en la actividad humana y no en el proceso informático. El único modo de conseguirlo es saber mandar al ordenador para que haga su trabajo. Para eso existe una comunicación verbal. Podemos decir que son palabras mágicas de un sirviente a nuestra disposición. Sólo así conseguiremos que el ordenador trabaje solo. No es que tenga actividad autónoma, sino que así automatizamos tareas.

C INMEDIATEZ

La comunicación lógica a través de órdenes es transmisión efectiva de mandatos. Esto también sucede en un entorno visual, pero tiene una variable importante. La imagen del escritorio es una simulación. Es una pequeña función representada

una calculadora binaria, sino a que es una máquina capaz de comunicación. Para lo que aquí nos interesa, capaz de recibir y emitir mensajes en un código previamente establecido.

Un ordenador es un asistente de trabajo. Para entender qué es la interacción lógica del ordenador, pensemos que estamos en una editorial. Nuestro asistente es una persona —es sólo una imagen— que no habla ningún idioma común con nosotros. Una tarea podría ser cortar papel. La única manera de conseguir que realice esa actividad es indicarle cómo se hace. Y luego guiarle para que lo repita. Es un ejemplo y soy consciente de que no existen ejemplos perfectos. Supongamos ahora que cambiamos de asistente y es una persona con conocimientos de inglés muy básico. El número de palabras que conoce es reducido. No le servirá para contar chistes o describir sus últimas vacaciones, pero su léxico le permite hacer su tarea. Conseguir que corte papel es pronunciar las palabras mágicas. Ya no hay que indicar paso a paso nada. Al ordenar, el asistente cumple eficazmente.

El ejemplo anterior en un ordenador sería un modelo de tarea simple. Mucha gente diría que se conseguría pinchando dos veces con el ratón sobre el botón pertinente. Totalmente de acuerdo. Lo relevante de un ordenador son las tareas complejas. Porque con ésas, un ordenador es todavía más útil que con las tareas simples. El ejemplo más básico de tarea compleja es la concatenación de tareas simples. Unir una tarea tras otra es muy útil para no tener que hacerlas individualmente.

Una concatenación de tareas simples puede ser convertir todos los archivos de un determinado tipo a otro formato. Además le cambiamos después el nombre a cada archivo nuevo. Nada impide que vayamos uno por uno, transformándolos a otro tipo de archivo y poniéndoles un nuevo nombre. Si son

V EL COSTE REAL

El método que propongo aquí tiene un precio exacto de cero euros. Los tres programas —`pandoc`, *Calibre* y ConTEXt— se desarrollan bajo una licencia de programación informática libre[15]. De los tres se permite descarga gratuita.

Desde Antonio Machado sabemos que es una necedad confundir valor con precio[16]. Hay un doble sentido en que podemos entender esto. En primer lugar, que algo se ofrezca gratuitamente no quiere decir que carezca de valor. No existen condiciones más o menos ocultas en la oferta de los programas. Todas están claras en sus respectivas licencias. En segundo lugar, un precio cero no supone necesariamente un coste cero. Porque nadie nace sabiendo y tenemos necesariamente que aprender.

A VALOR

El precio cero de un entorno informático puede ser una baza a su favor para que lo usemos. Ahora bien, no es la única variable que debemos de tener en cuenta. Si la informática es una herramienta, lo que debemos valorar es la utilidad de esos programas que nos ofrecen gratuitamente. Además, la gratuidad es sólo un aspecto de algo más grande: un desarrollo abierto. Dentro de ese desarrollo cooperativo podemos aportar ideas útiles al resto, cuando no directamente código.

El valor del método propuesto se tiene que analizar viendo cómo cumple el objetivo que se ha marcado. Se trata de un método de edición digital totalmente automatizado. Con este método, es posible conseguir resultados de alta calidad. Si

se prefiere, resultados profesionales. Completamente válidos para cualquier casa editorial que se precie. Tanto para un lector de libros electrónicos en formato digital, como directamente para su edición con la imprenta en formato de papel[17].

El método que describo supone un nuevo método de trabajo. Este nuevo método de trabajo no es sólo aprender a usar un nuevo programa. Por ejemplo, si quisiésemos usar *Adobe InDesign*, tendríamos que aprender primero cómo se usa. Nunca he usado *Adobe InDesign*, aunque intuyo que en la mayoría de los casos no exigirá un cambio de perspectiva en nuestro uso del ordenador.

Una de las virtudes de la edición digital automatizada que aquí describo es una nueva perspectiva de uso de ordenadores. Se trata de aprender a usar ordenadores, y no tanto programas. No es que no haya que saber usar los programas. Los programas son sólo herramientas dentro de la gran máquina que es el ordenador. Desde esa perspectiva, distintos programas pueden hacer de maneras parecidas distintas tareas. Como, por ejemplo, añadir formato a un texto, generar un sumario o incrustar una tipografía digital en un archivo.

La nueva perspectiva nos permite entender qué estamos haciendo antes de saber cómo hemos de hacerlo. Porque ese mucho más fácil entender cómo tenemos que hacer algo, si antes sabemos qué tenemos que hacer. Un ordenador es una máquina a la que mandamos, no a la que le pedimos. No tenemos que usar con ella extrañas artes de persuasión. Tampoco rogarle que tenga a bien hacer lo que queremos. Ni siquiera es cuestión de nuestro deseo. Se trata de saber qué tenemos que ordenar. Es la máquina la que tiene que someterse a nuestro imperio.

que usar así los ordenadores. Todo el error está en pensar que es la única forma de manejar un ordenador. Porque la imagen mental que construimos de cómo trabaja un ordenador es totalmente errónea. Y aunque nos parezca extraño —o incluso contradictorio—, un ordenador es mucho más interactivo de lo que es manejando un ratón en un entorno visual. Sin embargo, la metáfora de actividad de la imagen del entorno de escritorio es un modelo pobre de interacción.

B IMÁGENES Y PALABRAS

La última frase de la sección anterior nos puede parecer contradictoria. Voy a tratar de explicarla, porque en absoluto lo es. Usamos ordenadores para realizar tareas. Habrá muchas tareas simples. Aunque no todas las tareas serán simples, sino que las habrá complejas. Para evitar incomprensiones, complejo es lo que tiene partes y simple es lo que carece de ellas. Por contraposición, sencillo es lo que carece de artificio y complicado es lo rebuscado o artificioso. Simples o complejas son las cosas, sencillas o complicadas las hacemos las personas. Lo mismo se aplica a las tareas.

Automatizar tareas con ordenadores tiene dos motivos fundamentales: rapidez y fiabilidad. Un ordenador hace tareas que respondan a esquemas predeterminados mucho más rápido que cualquier persona. Y puede llegar a fallar, pero no se equivoca —de lo que no estamos exentos, como dice el refrán, quienes tenemos boca—. Eso pasa con las tareas simples y las complejas.

Un ordenador sólo habla. No cabe duda, que de una manera especial. No habla una lengua, sino que utiliza un código. El código permite una comunicación sin ambigüedades entre la persona y la máquina. En este sentido, un ordenador es una máquina lógica. Me refiero a que no tanto a que se trate de

objetos lógicos, se hacen más fácilmente inteligibles si se simula que son objetos físicos. El ejemplo más claro es una carpeta. La carpeta es la imagen de una clase de objeto —técnicamente se llama directorio— que puede contener otros objetos como archivos o más carpetas. En ningún lado, nada contiene nada físicamente. Ningún objeto está dentro de otro. Simplemente, la organización de los archivos hace que unos estén subordinados a otros. La imagen lógica es un organigrama jerárquico. Del mismo modo que nadie cobija a nadie en un organigrama jerárquico, funcionalmente un archivo está subordinado a otro.

También los iconos son un modo de visualizar que los archivos son algo lógico con apariencia física. Si no tuviésemos esa ayuda visual, nos podría costar más identificar archivos o carpetas. De este modo, la pantalla de un escritorio es la manera más fácil que tiene de manejarlo quien usa un ordenador. Porque es muy parecido a la misma manera que lo haría en una mesa con bolígrafos y papeles. O incluso, en una mesa con una máquina de escribir.

No cabe duda de que imaginarnos un entorno conocido ayuda a que podamos manejar más fácil y rápidamente un ordenador. Ahora bien, el entorno de escritorio es algo que el ordenador no necesita. Es una metáfora, es una simulación que esconde el modo que tiene un ordenador de recibir órdenes. Como máquina universal, un ordenador sólo ejecuta las órdenes que le dan. No puede hacer otra cosa. Si es una calculadora, las órdenes tienen que ser un texto de un tipo muy especial.

Por supuesto, antes de que alguien empiece con sudores fríos, conviene no olvidar algo: el entorno de escritorio es una simulación extremadamente útil. No digo que no tengamos

Aprender a editar digitalmente textos implica distinguir qué hace el ordenador y qué hace la persona. Eso supone qué partes de la tarea son automatizables y qué partes tiene que hacer la persona. La automatización ha de ser máxima. Ahora bien, está claro que la automatización no puede suplir la redacción del texto. Somos las personas quienes usamos las máquinas. Ninguno de los dos términos es prescindible. Por poner un ejemplo muy básico, el formato de una nota es algo que ha de hacer la máquina. Sin embargo, si no distinguimos la nota —del tipo que sea— del resto del texto, el ordenador no leerá nuestra cabecita. En un caso óptimo, puede que otra persona adivine cuál es la nota. Pero si no la distinguimos, el ordenador no es capaz de adivinar. Es una máquina de funcionamiento absolutamente simple: sigue órdenes y calcula.

B COSTE

Dice el refrán que todo lo que vale, cuesta. Aprender un nuevo modo de trabajar con un ordenador y de editar textos tiene un coste. Fundamentalmente de dos cosas, ya que hemos dejado el dinero de lado: tiempo y ganas. Todo aprendizaje es el paso de la ignorancia al saber. Para poder aprender, necesitamos tiempo y dedicación. Según los casos, será más o menos. Pero en todos los casos habremos de invertir tiempo en el aprendizaje de algo nuevo.

Esta advertencia es importante por varios motivos. Uno de ellos es que el cambio no es automático. La transición es inmediata cuando ya sabemos. O es automática para el ordenador, que no tiene nada que aprender. Si no sabemos, tendremos que dedicar un tiempo a ver cómo se hacen las cosas de otro modo. Nos llevará al menos dos o tres meses. Porque no todo nos funcionará a la primera. Necesitaremos tiempo para hacernos al cambio.

Del párrafo anterior se deduce que necesitamos un tiempo para experimientar. Si se quiere, para cacharrear. O sencillamente, para probar. Eso significa que no es buena idea tratar de aprender en el último momento. No debe hacerse para un proyecto cuya entrega sea inmediata. La negativa en ese caso es rotunta, porque nos arriesgamos a que ni el proyecto ni el aprendizaje lleguen a buen puerto. Como aprendizaje, supone una inviersión a medio plazo. Ése es el coste que debemos de calcular si nos compensa o no hacerlo.

El aprendizaje no sólo requiere tiempo. La sabiduría no se vislumbra con los años, sino con las muchas preguntas. El aprendizaje necesita de la inteligencia. Sin embargo este tipo de aprendizaje requiere, quizá más que otros, voluntad. La voluntad de la que hablo no es mero deseo, ganas, inclinación o incluso interés. Es sencilla y llanamente querer aprender y querer entender. Sí, cuando las cosas se pongan difíciles y haya que plantearlas de otra manera. No se trata de voluntarismo, de aprender y entender por pura fuerza de deseo, por puro mandato. Por pura orden, sólo memorizamos. La voluntad es la constancia que hace que el interés no pierda su norte. Que no dejemos de aprender por la dificultad, sino por la falta de utilidad.

C APRENDIZAJE

El aprendizaje exige tiempo y paciencia. Cuando aprendemos de adultos, lo hacemos por un interés mayor o más específico. La diferencia puede estar en que tenemos una cierta idea de qué queremos llegar a saber. O mejor dicho, tenemos un marco en el que encuadramos lo que queremos aprender. Todo aprendizaje es una búsqueda. Y la búsqueda necesita tiempo.

VI LA SIMULACIÓN VISUAL

Los ordenadores son calculadoras. Binarias, para más información. Probablemente, esa expresión no la entendamos la mayoría. No tenemos que saber cómo es un ordenador por dentro. En el fondo, supongo que será una cuestión de electrónica avanzada. Pero para usarlo con el mayor rendimiento posible, hemos de saber cómo trabaja un ordenador.

Cuando encendemos un ordenador, tenemos que esperar a que pueda estar interactivo. Para que lo podamos usar, tiene que pasar entre medio y un minuto. Es el transcurso que va desde que se enciende la pantalla hasta que el aparato está disponible. En ese tiempo, el ordenador carga lo que se llama el sistema operativo. Sin sistema operativo, un ordenador no funciona. Es más, es absolutamente incapaz de hacer nada. Desde luego, no puede hacer nada útil para que podamos usarlo.

Los sistemas operativos cargan también un entorno de escritorio como parte del sistema operativo. El entorno de escritorio es lo que nos deja una pantalla con más o menos iconos y ventanas cuando abrimos programas o carpetas que contienen archivos y otras carpetas. Propiamente, el entorno de escritorio no es necesario para que el sistema operativo funcione. Es extremadamente útil para que la persona que use el ordenador lo haga con mayor facilidad y comodidad.

A UNA METÁFORA DE ACTIVIDAD

El escritorio es la imagen visual de un escritorio de trabajo. Es una mesa en la que existen objetos. Si bien son realmente

rra Civil española, la Primera Guerra Mundial, la Revolución Francesa y la Revolución Rusa, nuestro relato de la historia contemporánea será una maravillosa fantasía.

El segundo ejemplo es el aprendizaje de los verbos irregulares en inglés. Es un ejercicio de pura memoria al que es fácil no verle el sentido. En realidad, no lo tiene de modo inmediato. Quien tiene que aprender, es que no sabe. Saber los verbos irregulares será relevante cuando la persona pueda hablar o escribir en inglés. La relevancia de su memorización se mostrará cuando los use en la lengua hablada o escrita. Ahora bien, la estructura del aprendizaje de una lengua es que el uso normalmente no será inmediato.

El aprendizaje a la carta suele ser un aprendizaje pobre. O el menú lo debemos configurar según nuestros intereses reales y no según la facilidad de aprendizaje. Si queremos aprender lo mínimo posible, nuestros resultados serán también paupérrimos. El aprendizaje requiere reconocer nuestra ignorancia para mediante el esfuerzo poder llegar a saber. Nadie puede sustituirnos en esa tarea exclusivamente nuestra.

La enseñanza de la informática en muchos casos es sencillamente adiestramiento en el uso de programas —los que sean—. Tiene una ventaja indiscutible: el tiempo necesario es menor. Sin embargo, el aprendizaje real es mucho más difícil. Dedicando un poco más de tiempo, alcanzaremos un nivel de uso experto del ordenador. Desde ese nivel, aprender cualquier otro programa será entonces mucho más fácil. Porque habremos entendido cómo funciona un ordenador. No a nivel físico de electrónica, sino a nivel lógico de procesamiento de órdenes.

Una dedicación de aprendizaje sin un resultado claro nos puede parecer una pérdida de tiempo. Aunque si el resultado está ya dado de antemano, nuestro aprendizaje será relativo porque no partimos del desconocimiento. Invertir tiempo es algo necesario. Tan ineludible, que si no tenemos tiempo de aprender, es mejor que nos olvidemos de la edición digital automatizada. Porque no nos servirá para nada. Los conocimientos son útiles cuando se poseen. O en el mejor de los casos, cuando conocemos a quienes los poseen y podemos convencerles de que hagan nuestra tarea.

El tiempo necesario lleva consigo una fuerte dosis de paciencia. Por supuesto, variará según el tipo de persona que sea cada cual. Pero el proceso de aprendizaje tiene un componente muy básico en el ensayo y error. Esto es, probar a ver si funciona. En la mayoría de los casos, funcionará. Aunque en algunos no. Y puede llegar a ser exasperante. Porque un ordenador es una máquina calculadora. Puede llegar a fallar en casos extraordinarios, pero no se equivoca nunca. Cumple escrupulosamente lo que le dicen. No nos va a decir qué le estamos ordenando mal. En algunos casos, puede ser desesperante. En esos momentos, además de paciencia, probablemente necesitamos un descanso.

Pasar de lo desconocido a lo conocido es un proceso. No es algo que nos suceda sin más. Es más bien algo que hacemos, como seguir un camino. La particularidad es que a veces ese camino no estará marcado. El aprendizaje es también una búsqueda. E igual que el aprendizaje es personal —quien pueda llegar a enseñar no nos suplanta en la tarea de aprender—, la búsqueda es personal. Porque nadie puede recorrer nuestro camino, lo hemos de recorrer cada cual.

Toda búsqueda incluye preguntas. Preguntar es preguntarnos. En muchos casos sobre cosas que creemos entender, pero que realmente no comprendemos. También podemos preguntar a otras personas. Preguntar a otras personas es pedir ayuda. Cuando pedimos ayuda no debemos olvidar que no dejamos de aprender nosotros. La ayuda nos tiene que servir para el aprendizaje. Si nos resuelven los problemas otras personas, pasamos a tener nosotros el problema de no haber aprendido nada. Pedir ayuda no es desentendernos de los problemas que se nos presentan. Es buscar otro modo de afrontarlos y de tratar de resolverlos.

El aprendizaje no es una actividad puntual, que se hace durante un momento o un tiempo y ya está. Dejaremos de aprender cuando descansemos en la eternidad. Por tanto, la paciencia también requiere no pretender saber todo de una vez para siempre. El camino es largo, pero no necesitamos partir desde la meta. Con paciencia, iremos aprendiendo en la medida en que vayamos necesitándolo. Y sólo así podremos integrar este aprendizaje con el resto de nuestros conocimientos.

D ESFUERZO

El aprendizaje es imposible sin esfuerzo. Por puro interés no aprendemos. O sólo aprendemos lo que es fácil. El esfuerzo es necesario para asimilar lo que nos es extraño. El aprendizaje no existe sin resistencia, además de nuestro impulso algunas veces contrario. Vencer esa resistencia implica esforzarse por aprender. Esfuerzo no es sufrimiento, aunque hayamos vinculado el padecimiento al aprendizaje desde la Grecia clásica [18].

El esfuerzo no es necesario porque lo que tenemos que aprender sea extremadamente complicado. Es porque se trata de algo nuevo. Y además tenemos que hacernos con una nueva perspectiva. Nos puede costar más o menos, pero no debemos dejar de aprender porque nos cueste. Sin esfuerzo sólo aprenderemos obviedades. Realmente no nos cultivaremos, ni nos libraremos de nuestra ignorancia. El esfuerzo es necesario para superar el reto que supone el aprendizaje. Ese pequeño sacrificio merece la pena por el resultado de nuestro mayor conocimiento.

En en el fondo, todo se basa en nuestra decisión personal de aprender. Esa determinación es absolutamente intransferible. Entenderemos el sentido de todo cuando ya sepamos. El peligro es juzgar que algo es relevante o no desde lo que es estrictamente desconocimiento. Cuando aprendemos es muy difícil valorar qué conocimientos son relevantes y cuáles no. Sobre todo si confundimos la relevancia con la facilidad de aprendizaje. Siempre podemos volver sobre lo que necesitemos. Pero si evitamos afrontar las mayores dificultades, nuestro proceso de aprendizaje se eternizará.

Pongo unos ejemplos que no tienen nada que ver con la informática. El primero es sobre las fechas y la historia. Por supuesto que la historia no son sólo fechas. Saber historia mundial no es saber hechos y fechas, sino poder explicar una época de la humanidad. Ahora bien, si no sabemos ordenar estos acontecimientos: la Segunda Guerra Mundial, la Gue-